monatlich

Compute mit

COMMODORE & SCHNEIDER

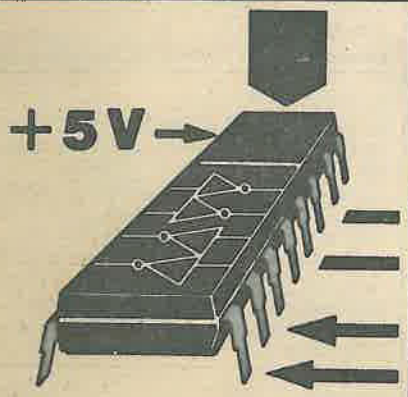
7/86

VC-64, VC-20, C-16/116, CPC-464

Unabhängiges Magazin für Anwender von Commodore- und Schneider-Computern

3,80 DM
33 öS
3,80 sFr

Aktuell +++ Aktuell +++ Aktuell +++ Aktuell



64 KB für C-16/116 im Selbstbau

Seite 38

Commodore 64: Basic Extension

bindet Ihre Assemblerroutinen mit einem Befehlswort in das Basic ein. Schicken Sie uns Ihre Assemblerroutinen, für eine „Compute mit"-Bibliothek, damit sich unsere Leser ihr ganz persönliches Basic schaffen können!

Seite 22

Der neue Commodore 64 im Test!

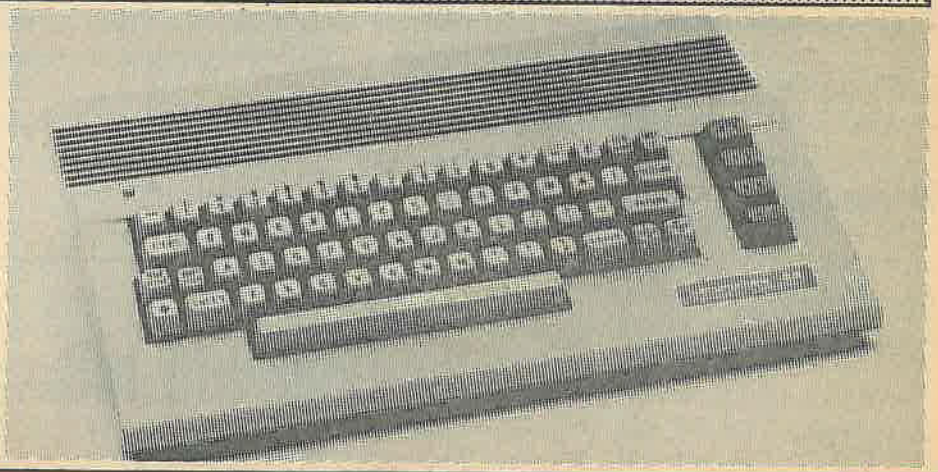
Commodore stellte im Mai den neuen C-64 vor.

– Was ist eigentlich neu, außer dem Gehäuse?

Dem neuen C-64 legt Commodore ein neues Betriebssystem bei.

– Wie leistungsfähig kann ein Betriebssystem mit Fenstertechnik auf dem C-64 sein?

Seite 3



Kleinanzeigen – Report – Software-Abo – Software-Service
Bücher – Werkstatt – Assembler-Kurs – Software-Reviews – Tips & Tricks
Schneider-Disketten-Archiv – Roadrunner – Break-Out – u. v. m.

inhalt

Report

Der neue C-64 3

Bericht: Jugend forscht '86 4

Computer Camp 4

Was gibt es Neues auf dem Softwaremarkt 20

Bücher

Z-80 Maschinensprache 54

DFÜ für C-64 & 128 54

Der Computerdieb 55

Schneider-Floppy-Buch 55

Tips & Tricks

Scrolling für VC-20 10

INPUT und READ ERROR (C-64) 26

UPDATE für BEEP-Routine (C-16/116) 37

Software-Service 32

Kleinanzeigen 59

Assemblerkurs 44
Teil 14

Werkstatt:

VC-20 Sound-Demo 12

Speichererweiterung für C-16/116 38

Diskettenarchiv für CPC 464, 664 und 6128 51

Software

Commodore

Mad Rush (VC-20) 5

Ghost-Hell (VC-20) 8

Roadrunner (C-64) 13

Basic Extension (C-64) 22

Meteoritensturm (C-64) 26

Checksummer OC für C-16/116 31

Airwolf (C-16/116) 34

Break-Out (C-16/116) 41

Schneider

Poker 46

Cave-Walker 57

Korrekturen

Klartext für C-16/116 30

Fehlerteufel 59

Die nächste Ausgabe von

Compute mit

liegt wieder ab

16. Juli

bei Ihrem Zeitschriftenhändler für Sie bereit.

Impressum

„Compute mit"
erscheint monatlich im Tronic-Verlag, 3440 Eschwege

Redaktion:
Axel Credé (verantwortlich)
Chefredakteur: Hartmut Wendt
Uwe Knierim, Manfred Kleimann, Bernd Zimmermann
Frank Brall, Ottfried Schmidt, Siegfried Görk

Gesamtherstellung:
Druckhaus Dierichs Kassel, Frankfurter Str. 168, 3500 Kassel

Vertrieb:
Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz
Verlagsunion
Friedrich-Bergius-Straße 20
6200 Wiesbaden
Telefon 06121/2660

Anfragen nicht an Vertrieb oder Druckerei, sondern nur an den Verlag!

Erscheinungsweise:
Erstverkaufstag von „Compute mit" jeweils Mitte des Monats.

Urheberrecht:
Alle in „Compute mit" veröffentlichten Beiträge sind urheberrechtlich geschützt. Alle Rechte, auch Übersetzungen, vorbehalten.

Reproduktionen jeder Art (Fotokopien, Microfilm, Erfassung in Datenverarbeitungsanlagen usw.) bedürfen der schriftlichen Genehmigung des Verlags. Alle veröffentlichte Software wurde von Mitarbeitern des Verlages oder von freien Mitarbeitern erstellt.

Aus ihrer Veröffentlichung kann nicht geschlossen werden, daß die beschriebenen Lösungen oder Bezeichnungen frei von Schutzrechten sind.

Bezugspreis:
Einzelheft 3,80 DM
Abonnement: Inland 42,– DM im Jahr (12 Ausgaben)
Ausland: Europa 52,– DM
ohne Kassetten!

Autoren, Manuskripte:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen.

Sollte keine andere Vereinbarung getroffen sein, so gehen wir davon aus, daß Sie mit einem Honorar von 120,– DM pro abgedruckter Seite im Heft einverstanden sind.

Bei Zusendung von Manuskripten und Software erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger.

Rücksendung erfolgt nur gegen Erstattung der Kosten. Zusendungen von Software zur Veröffentlichung sollten folgendes enthalten:
Kopierfähige Kassette oder Diskette mit dem Programm (Computer-Bezeichnung), von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinenlistings), evtl. Bildschirmfotos von einem Probelauf und ausführliche Programmbeschreibung (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaues, Farbe, Grafik usw.). Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden.

Anzeigenpreise:
Bitte Mediaunterlagen anfordern.

Anzeigenverwaltung:
Tronic-Verlag GmbH,
Am Stad 35, 3440 Eschwege
Telefon 05651/30011
Telefax 05651/30014

Anzeigenleitung:
Inland: H. Wendt
Ausland: M. Kleimann

COMMODORE-NEUHEITEN

Endlich ist das Geheimnis gelüftet: Der neue C-64 ist da!

Überraschend hatte Commodore eine Veränderung des C-64 angekündigt. Neugierig haben wir uns gefragt, worin nun die eigentlichen Veränderungen liegen. Der erste Eindruck: der C-64 ist schlanker geworden. Das neue Design erinnert etwas an den PC 128, jedoch ohne 10'er-Tastatur. Das alte Gehäuse war ja häufiger Anlaß für Spott. Butterbrotschachtel hieß es, oder man sprach liebevoll von dem Käfereffekt. Nun sind die Zeiten vorbei: der C-64 hat sich das gängige „Slim-line"-Design der heutigen Computergeneration zu eigen gemacht. Allerdings: ganz so ergonomisch wie der PC 128 ist der neue C-64 nicht. Die Handballen müssen sich noch immer etwas krümmen, es fehlt vorne etwas Länge und die Tastatur ist immer noch etwas schräg, kleine Reminiscensen an den alten C-64!

Gibt es aber technische Neuerungen, ist die nächte Frage. Schraubt man das Gehäuse auf, wird man von einer wunderschönen Ohriginalplatine angelacht, wie wir sie bisher von allen unseren Oldtimermodellen her gewöhnt sind. Lediglich der VIC-Chip und der I/O-6526 tragen neue Seriennummern, ohne jedoch auf eine entschiedene Veränderung schließen zu lassen. Die einzige wirkliche Veränderung besteht in der wesentlich verbesserten Abschirmung der Platine. Weitere Veränderungen wurden nicht vorgenommen, um die Kompatilität zu den alten Geräten 100 %ig zu gewährleisten.

Tatsächlich aber vermissen wir kleine Verbesserungen in Bereichen, die die Kompatilität in keiner Weise gefährdet hätten. So muß sich der User nach wie vor einen Resettaster selbst stricken. Auch eine Betriebssystemumschaltplatine hätte integriert sein können. Der obligatorische Eigenbau bleibt auch hier nicht erspart, was jedoch durch das neue Gehäusekonzept erheblich erschwert wird. Der Raum ist so eng, daß für eine Umschaltplatine zumindest ein Teil des Kühlblechs herausgesägt werden muß.

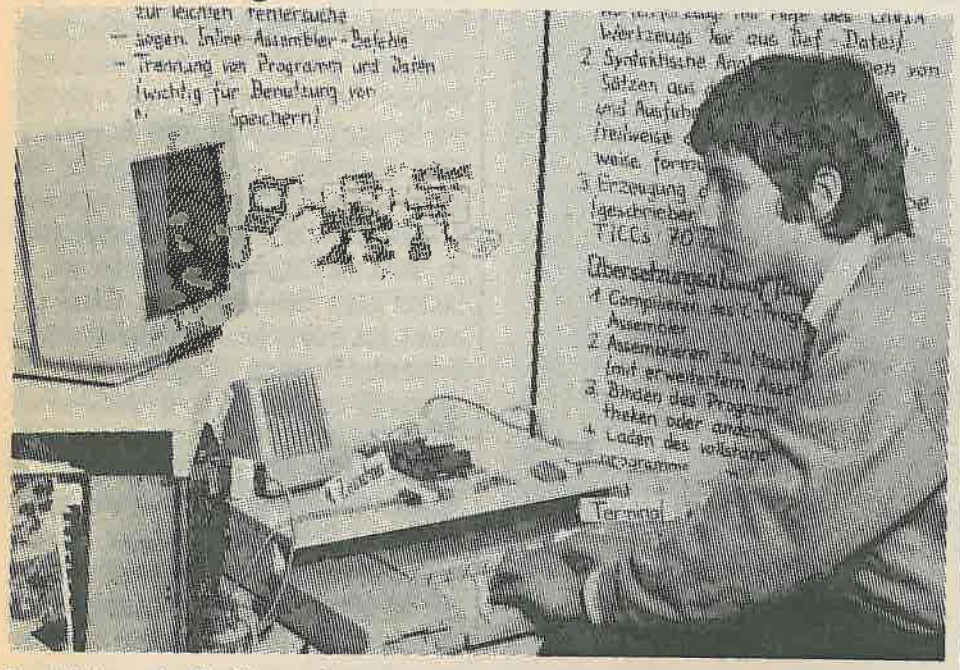
Eine weitere wichtige Neuerung für den C-64-Käufer besteht in einer besonderen Beigabe. Commodore hat nun endgültig die Rechte an einem neuen Betriebssystem erworben, das erstmalig auf der letztjährigen CES-Messe in Las Vegas vorgestellt wurde, und legt dieses jedem neuen Gerät als Zugabe auf Diskette bei.

Dieses neue Betriebssystem, GEOS genannt, hat 2 entscheidende Merkmale. Es bietet dem Benutzer eine grafische Benutzeroberfläche ähnlich dem AMIGA: Pull-Down-Menü mit Piktogrammen und Softwareauswahl per Joystick und Maus. Zweiter Vorteil von GEOS ist ein implementierter Floppybeschleuniger, der ungefähr 10-fache Geschwindigkeit erreicht. Der ist allerdings auch unbedingt erforderlich, da das Betriebssystem in Modulen vorliegt, die immer wieder von der Betriebssystemdiskette nachgeladen werden müssen. Tatsächlich bekommt der C-64 mit GEOS etwas von den Vorteilen der 16-Bitter eingehaucht und wird auch in einigen Funktionen erstaunlich schnell (vor allem wenn es darum geht, Grafik nachzuladen). Einige Nachteile sollen nicht verschwiegen werden: innerhalb von GEOS ist kein Basic-Editor integriert. Man muß GEOS verlassen, wenn man ins Basic gelangen will, kann jedoch durch einen einfachen Druck auf die RESTORE-Taste wieder in das GEOS gelangen. Ferner ist bekannt, daß die ersten Geräte nur mit der amerikanischen Version von GEOS ausgerüstet werden. Die deutsche Version ist noch in Arbeit. Insgesamt jedoch ist GEOS eine fantastische Erweiterung, und es kommt jetzt darauf an, daß genügend Pro-

grammierer sich damit beschäftigen – ein GEOS-Basic wäre dann auch denkbar. Auf der GEOS-Diskette befinden sich schon 2 GEOS-Programme: GEOwrite und GEOpaint, über die wir das nächste Mal berichten werden.

Als weitere Neuheiten kündigt Commodore sowohl für den C-64 als auch für den PC 128 Speichererweiterungen an. Der C-64 soll auf 512-KB und der PC 128 auf 640 KB aufgerüstet werden können.

JUGEND FORSCHT: Sieger ermittelt



Pascal Merle vor seiner Einrichtung zur Programmierung eines Zielprozessors (bester JUGEND-FORSCHT-Teilnehmer im Bereich Informatik)
Foto: Ford-Werke AG

Vom 8. bis 14. Mai fand der **21. Bundeswettbewerb JUGEND FORSCHT** in Saarlouis statt. Am Montag wurden bereits die Preisträger bekanntgegeben. Auffallend bei der Preisvergabe war die Vielzahl der Arbeiten, die sich des Computers als wissenschaftliches Hilfsmittel bedienten. Das beste Beispiel hierfür ist wohl die Arbeit des Bundessiegers für Biologie, Thomas Dick (17). In einer Computersimulation entwickelte er den Aufbau eines Enzyms, über dessen Molekularstruktur noch wenig bekannt ist. Für diese Arbeit, die bereits internationale wissenschaftliche Anerkennung gefunden hat, erhielt Thomas außerdem den Sonderpreis des Bundespräsidenten. Auch bei vielen anderen Arbeiten im Bereich Physik und Chemie spielten Computer eine herausragende Rolle. Um so erstaunlicher war es, daß im Bereich Mathematik/Informatik kein erster Preis vergeben wurde. Den 2. Preis erhielt der erst 14jährige Pascal Merle für ein Computerprogramm in der höheren Programmiersprache „C", das ein anderes C-Programm in Maschinensprache übersetzt, die ein bestimmter Zielprozessor (TMS 99xxx) versteht.

Viele Arbeiten beschäftigten sich auch mit unserer natürlichen Umwelt und deren Schädigung. Dies veranlaßte Ford-Generaldirektor Daniel Goeudevert zu der Bemerkung: „Ich verstehe auch, daß viele junge Leute vom Aussterben ganzer Tiergattungen zum Beispiel mehr beunruhigt sind, als sie vom technologischen Fortschritt begeistert sind. Diesen Wertewandel müssen wir in unserer Produktentwicklung berücksichtigen. Denn technischer Fortschritt ist kein Selbstzweck, sondern dient stets den menschlichen Bedürfnissen und den gesellschaftlichen Notwendigkeiten."

Wir schließen uns an dieser Stelle den zahlreichen Gratulanten an und wünschen den Siegern alles Gute für Ihre Zukunft.

Sommerferien im Computercamp

Für alle, die ihre Sommerferien mit einem Computerkurs verknüpfen wollen, bietet COMPUCAMP wieder Ferien im Computercamp an.

Das Angebot richtet sich an Jugendliche und auch an Kinder. Der Akzent dieses Angebots liegt dabei nicht einseitig auf „Lernen", auch die Freizeit soll dabei nicht zu kurz kommen. Nicht Bildschirmfanatiker sind also gefragt, sondern alle, die auch Spaß an Windsurfen, Wasserski und Tennis haben. Zum Programm gehören auch Ausflüge zu Firmen wie COMMODORE und Fischertechnik.

Die Computerkurse sind weit gestaffelt – von LOGO, über Basic bis zur Maschinensprache. Für jeden wird also etwas passendes angeboten. Camp finden statt in

Schloß Dankern (Emsland), Tönning (Nordsee) und Veltishof am Titisee im Schwarzwald.

CompuCamp GmbH, Goßlerstr. 21, 2000 Hamburg 55, Tel.: 040/861255

Schnelles Radrennen für den VC 20

MAD RUSH

Dies ist ein wirklich schnelles Spiel für den VC 20

In bester Formel 1 - Art muß ein Radfahrer durch einen Straßenkurs gesteuert werden, ohne gegen die auftauchenden Hindernisse zu stoßen: Bäume, Ölfässer, Kakteen, Palmen und Sträucher. Außerdem kann man auf Ölpfützen zur Seite wegrutschen. Das Spiel wird laufend schneller – wer 30 Meilen schafft ist schon ein Rennprofi!

Auf geht's!

Zum Programmaufbau:

Das Programm besteht aus 5 Vorprogrammen und einem Hauptprogramm. Nachdem alle Vorprogramme einzeln abgetippt und abgespeichert wurden, gibt es die Möglichkeit, sie zu einem einzigen Vorprogramm zusammenzufassen:

1. Man lädt alle Vorprogramme einzeln und startet auch jedes für sich.
2. Dann gibt man ein: POKE 45,230: POKE 46,29: CLR
3. Man speichert es nun als <u>ein</u> Vorprogramm ab.

Beim Abtippen ist darauf zu achten, daß keine Veränderungen am Programm vorgenommen werden, da das Programm sonst aussteigen kann.

```
0 REM*****************                    <115>
1 REM*    MAD RUSH    *                   <38>
2 REM*                *                   <99>
3 REM*     1986 BY    *                   <201>
4 REM*  HEINZ KROOS   *                   <69>
5 REM* NEUE REIHE 26 *                    <77>
6 REM*    4795 BOKE   *                   <37>
7 REM*****************                    <122>
9 PRINTCHR$(14)"{CLEAR SPACE6}\AD-_USH":
PRINT"{SPACE6}━━━━━━━━━━━":PRINT"{DOWN3 SPA
CE4}-ATA$(SPACE)READING"                  <61>
10 FORA=7000TO7679:READB:POKEA,B:C=C+B:N
EXT:PRINTCHR$(142)"{CLEAR}"               <46>
11 IFC<>61957THENPRINT"DATA{SPACE}ERROR"
:END                                      <255>
12 POKE198,1:POKE631,131                  <21>
100 DATA162,0,189,0,31,157,22,31          <209>
101 DATA189,0,151,157,22,151,202,208      <32>
102 DATA241,162,0,189,0,30,157,22         <251>
103 DATA30,189,0,150,157,22,150,202       <36>
104 DATA208,241,189,255,29,141,44,30      <203>
105 DATA189,252,29,141,44,151,189,253     <132>
106 DATA29,141,44,150,189,254,29,141      <93>
107 DATA44,31,76,149,26,2,0,0             <167>
108 DATA1,0,32,1,216,0,15,0               <249>
109 DATA0,0,0,0,0,0,0,0                   <68>
110 DATA0,0,0,0,0,0,0,0                   <69>
111 DATA0,0,0,0,0,0,0,0                   <70>
112 DATA0,0,0,0,96,96,96,96               <141>
113 DATA96,96,96,96,96,96,96,96           <164>
114 DATA96,96,96,96,96,96,96,96           <165>
115 DATA98,97,96,96,96,96,96,96           <178>
116 DATA96,96,96,96,96,96,96,96           <167>
117 DATA96,96,96,96,96,96,96,96           <168>
118 DATA96,96,96,96,96,96,4,96            <60>
119 DATA98,97,96,96,96,96,96,96           <182>
120 DATA96,96,96,96,96,96,96,140          <115>
121 DATA16,16,16,112,156,178,178,186      <176>
122 DATA254,124,56,24,24,16,16,16         <94>
123 DATA8,8,8,14,57,77,77,93              <196>
124 DATA127,62,28,24,24,8,8,8             <5>
125 DATA12,26,58,170,234,202,108,120      <229>
126 DATA120,58,29,13,13,13,5,2            <43>
127 DATA227,211,203,199,203,211,227,227   <215>
128 DATA255,255,255,255,255,255,255,255   <111>
129 DATA24,60,126,254,254,127,127,14      <246>
130 DATA60,126,255,255,254,124,124,48     <52>
131 DATA32,88,92,85,87,83,54,30           <144>
132 DATA30,92,184,176,176,176,160,64      <25>
133 DATA26,79,181,174,87,189,82,47        <252>
134 DATA224,80,168,104,164,216,116,168    <118>
135 DATA13,7,7,7,7,15,31,41               <245>
136 DATA176,224,192,192,192,224,240,88    <201>
137 DATA60,60,60,189,255,255,255,255      <62>
138 DATA189,60,60,60,60,60,60,126         <47>
139 DATA7,24,32,56,63,63,59,53            <226>
140 DATA224,24,4,28,252,252,220,92        <187>
141 DATA53,53,53,59,63,63,31,7            <103>
142 DATA92,92,92,76,124,252,248,224       <246>
143 DATA88,120,126,30,26,24,24,24         <247>
144 DATA48,56,248,248,251,255,191,63      <0>
145 DATA63,56,56,56,56,56,56,56           <45>
146 DATA0,0,0,0,0,0,48,127                <239>
147 DATA99,159,254,195,127,124,7,0        <233>
148 DATA240,159,255,49,255,135,255,120    <27>
149 DATA0,0,240,255,143,248,207,252       <148>
150 DATA0,0,0,0,240,255,248,143           <190>
151 DATA0,0,0,0,0,0,240,158               <175>
152 DATA7,0,0,0,0,0,0,0                   <125>
153 DATA0,0,0,0,0,0,0,0                   <112>
154 DATA240,127,7,0,0,0,0,0               <139>
155 DATA0,0,0,0,0,0,0,0                   <114>
156 DATA223,253,199,127,7,0,0,0           <211>
157 DATA253,29,249,202,250,124,0,0        <108>
158 DATA1,7,5,26,111,90,245,174           <21>
159 DATA192,120,180,236,84,234,85,174     <231>
160 DATA173,82,61,86,93,181,106,13        <20>
161 DATA90,244,90,173,234,168,80,160      <110>
162 DATA15,15,7,7,7,7,7,7                 <43>
163 DATA224,224,192,192,192,192,192,192   <92>
164 DATA7,7,15,15,31,63,82,0              <14>
165 DATA192,192,224,224,240,248,150,0     <243>
166 DATA53,19,73,54,75,37,31,1            <27>
167 DATA180,168,114,212,168,118,216,224   <58>
168 DATA0,0,0,2,10,21,19,75               <27>
169 DATA0,0,0,64,208,80,104,240           <60>
170 DATA16,8,6,114,31,15,7,15             <104>
171 DATA0,12,48,96,108,216,240,224        <251>
172 DATA25,48,32,64,0,0,0,0               <254>
173 DATA128,192,96,96,48,48,48,24         <112>
```

```
174 DATA24,24,24,24,24,24,60,126          <72>
175 DATA1,3,1,3,7,1,3,7                   <158>
176 DATA0,128,0,128,192,0,128,192         <245>
177 DATA15,3,7,15,31,7,15,31              <254>
178 DATA224,128,192,224,240,192,224,240   <173>
179 DATA63,15,31,63,127,3,3,3             <25>
180 DATA248,224,240,248,252,192,192,192   <39>
181 DATA00,0,0,0,0,0,0,0                  <160>
182 DATA0,0,0,0,198,229,238,101           <223>
183 DATA0,255,0,255,32,53,49,32,53,14,32
,53,49,32,53,14                           <14>
```

```
0 REM*****************                    <115>
1 REM*                 *                  <98>
2 REM*   MAD RUSH      *                  <39>
3 REM*                 *                  <100>
4 REM*    TEIL 2       *                  <218>
5 REM*                 *                  <102>
6 REM*****************                    <121>
9 PRINTCHR$(14)"{CLEAR SPACE7}\AD-_USH":
PRINT"{SPACE7}━━━━━━━━━":PRINT"{DOWN3 SPA
CE4}-ATAS{SPACE}READING"                  <219>
10 FORA=6600TO6999:READB:POKEA,B:C=C+B:N
EXT                                       <29>
11 PRINT"{CLEAR}"CHR$(142):IFC<>43189THE
NPRINT"DATA{SPACE}ERROR":END              <84>
12 POKE198,1:POKE631,131                  <21>
100 DATA 169,127,141,34,145,189,32,145    <60>
101 DATA 75,128,201,64,240,3,76,89        <107>
102 DATA 26,76,169,26,96,169,6,141        <153>
103 DATA 4,30,169,6,141,14,30,169         <153>
104 DATA 0,141,4,150,169,0,141,14         <176>
105 DATA 150,173,49,24,141,224,25,173     <200>
106 DATA 49,24,105,9,141,229,25,255       <54>
107 DATA 248,25,255,242,25,173,49,24      <192>
108 DATA 141,234,25,173,49,24,105,9       <14>
109 DATA 141,239,25,255,6,26,255,12       <212>
110 DATA 26,76,238,26,96,169,0,141        <0>
111 DATA 28,27,96,96,96,96,96,96          <125>
112 DATA 96,96,96,96,169,4,141,150        <16>
113 DATA 27,255,151,27,173,151,27,41      <21>
114 DATA 2,240,23,206,155,26,206,185      <152>
115 DATA 26,206,193,26,206,203,26,206     <134>
116 DATA 198,26,169,0,141,151,27,206      <0>
117 DATA 152,26,76,181,26,96,96,96        <194>
118 DATA 96,169,10,141,150,27,255,152     <188>
119 DATA 27,173,152,27,41,2,240,28        <43>
120 DATA 255,155,26,255,185,26,255,193    <12>
121 DATA 26,255,203,26,255,198,26,169     <144>
122 DATA 0,141,152,27,255,152,26,169      <17>
123 DATA 254,141,191,26,76,181,26,169     <33>
124 DATA 0,141,147,27,169,25,141,148      <232>
125 DATA 27,76,166,25,96,169,32,141       <134>
126 DATA 239,31,189,217,31,201,32,240     <231>
127 DATA 3,76,100,25,76,200,25,96         <211>
128 DATA 96,189,17,145,75,16,201,8        <220>
129 DATA 240,3,76,44,26,173,150,27        <189>
130 DATA 141,217,31,173,150,27,233,255    <190>
131 DATA 141,239,31,169,0,141,217,151     <115>
132 DATA 169,0,141,239,151,255,150,27     <244>
133 DATA 255,150,27,189,150,27,41,4       <197>
134 DATA 240,5,169,0,141,150,27,169       <55>
135 DATA 255,141,34,145,169,255,141,191   <86>
136 DATA 26,76,0,25,96,96,173,26          <194>
137 DATA 27,201,32,240,7,173,0,30         <112>
138 DATA 201,6,240,3,76,7,27,169          <146>
139 DATA 32,141,26,27,76,30,27,173        <243>
140 DATA 0,30,201,6,240,3,76,25           <205>
141 DATA 27,169,7,141,26,27,76,30         <222>
142 DATA 27,169,7,141,0,30,255,246        <163>
143 DATA 26,255,8,27,255,28,27,173        <149>
144 DATA 246,26,201,22,240,3,76,238       <240>
145 DATA 26,169,0,141,246,26,169,0        <196>
146 DATA 141,8,27,76,29,26,96,96          <157>
147 DATA 189,22,150,141,253,29,189,22     <130>
148 DATA 151,141,252,29,189,22,30,141     <142>
149 DATA 255,29,189,22,31,141,254,29      <185>
  ENDE DES LISTINGS
```

```
0 REM@@@@@@@@@@@@@@@@@@                   <15>
1 REM@                 @                  <208>
2 REM@   MAD-RUSH      @                  <191>
3 REM@                 @                  <210>
4 REM@    TEIL 3       @                  <189>
5 REM@                 @                  <212>
6 REM@@@@@@@@@@@@@@@@@@                   <21>
9 PRINTCHR$(14)"{CLEAR SPACE6}\AD-_USH":
PRINT"{SPACE6}━━━━━━━━━":PRINT"{DOWN3 SPA
CE4}-ATAS{SPACE}READING"                  <61>
10 FORA=6300TO6599:READB:POKEA,B:C=C+B:N
EXT                                       <230>
11 PRINT"{CLEAR}"CHR$(142):IFC<>21418THE
NPRINT"DATA{SPACE}ERROR":END              <175>
12 POKE198,1:POKE631,131                  <21>
100 DATA 6,6,6,5,5,5,5,4                  <159>
101 DATA 4,4,4,4,4,5,5,5                  <134>
102 DATA 6,6,6,6,6,6,6,6                  <237>
103 DATA 7,7,8,8,8,8,8,8                  <120>
104 DATA 9,9,9,8,8,8,8,7                  <123>
105 DATA 7,7,7,6,5,5,5,5                  <200>
106 DATA 5,5,5,5,5,5,5,5                  <169>
107 DATA 5,5,5,5,5,5,5,5                  <170>
108 DATA 5,5,5,5,5,5,5,5                  <171>
109 DATA 5,5,5,5,5,5,5,5                  <172>
110 DATA 5,5,5,5,5,5,5,5                  <173>
111 DATA 5,5,5,5,5,5,5,5                  <174>
112 DATA 5,5,5,5,169,32,141,0             <191>
113 DATA 30,169,5,141,0,150,255,3         <216>
114 DATA 25,255,8,25,173,3,25,201         <238>
115 DATA 22,240,3,76,0,25,169,0           <183>
116 DATA 141,3,25,169,0,141,8,25          <114>
117 DATA 76,221,25,173,77,23,141,11       <42>
118 DATA 144,173,77,23,141,12,144,206     <80>
119 DATA 158,27,173,158,27,201,0,240      <97>
120 DATA 3,76,80,25,206,46,25,169         <95>
121 DATA 15,141,158,27,169,17,141,14      <98>
122 DATA 144,206,40,25,206,14,144,76      <83>
123 DATA 191,234,96,96,96,96,96,96        <171>
124 DATA 96,96,96,96,96,96,96,96          <175>
125 DATA 173,155,26,141,107,25,173,221    <49>
126 DATA 31,141,154,27,173,154,27,201     <137>
127 DATA 8,240,3,76,137,25,255,155        <60>
128 DATA 26,255,185,26,255,193,26,255     <17>
129 DATA 198,26,76,153,22,173,154,27      <194>
130 DATA 201,9,240,3,96,96,96,206         <186>
131 DATA 155,26,206,185,26,206,193,26     <140>
132 DATA 206,198,26,76,146,22,96,96       <174>
133 DATA 96,96,169,0,201,22,240,9         <46>
134 DATA 255,167,25,32,64,27,76,166       <7>
135 DATA 25,169,0,141,167,25,169,149      <151>
136 DATA 141,147,27,169,26,141,148,27     <191>
137 DATA 96,96,96,96                      <198>
  ENDE DES LISTINGS
```

```
0 REM*    TEIL4      *                                          <228>
9 PRINTCHR$(14)"{CLEAR SPACE6}\AD-_USH":
PRINT"{SPACE6}━━━━━━━━━━":PRINT"{DOWN3 SPA
CE4}-ATAS{SPACE}READING"                                        <61>
10 FORA=5765TO5972:READB:POKEA,B:C=C+B:N
EXT                                                             <242>
11 FORA=6130TO6299:READB:POKEA,B:C=C+B:N
EXT                                                             <209>
12 PRINT"{CLEAR}"CHR$(142):IFC<>36910THE
NPRINT"DATA{SPACE}ERROR":END                                    <207>
13 POKE198,1:POKE631,131                                        <22>
100 DATA173,224,25,141,238,27,255,255                           <101>
101 DATA27,76,64,27,96,206,203,26                               <215>
102 DATA206,152,26,96,255,203,26,255                            <91>
103 DATA152,26,96,96,169,0,141,14                               <157>
104 DATA144,169,191,141,20,3,169,234                            <221>
105 DATA141,21,3,96,169,39,141,20                               <130>
106 DATA3,169,25,141,21,3,169,255                               <231>
107 DATA141,40,25,160,255,141,46,25                             <255>
108 DATA96,173,155,26,105,22,141,152                            <28>
109 DATA26,76,135,26,96,173,224,25                              <91>
110 DATA105,210,141,156,27,173,156,27                           <43>
111 DATA141,155,26,173,156,27,141,185                           <88>
112 DATA26,173,156,27,141,198,26,173                            <221>
113 DATA156,27,105,22,141,193,26,173                            <245>
114 DATA156,27,105,22,141,203,26,76                             <12>
115 DATA198,22,96,227,223,227,223,227                           <22>
116 DATA230,231,234,236,238,239,230,230                         <147>
117 DATA234,230,234,231,231,230,230,227                         <169>
118 DATA227,223,223,0,0,223,234,239                             <195>
119 DATA238,239,236,238,236,239,234,238                         <118>
120 DATA234,236,234,239,231,238,231,236                         <55>
121 DATA231,234,231,239,230,238,230,236                         <200>
122 DATA230,234,230,231,230,239,227,238                         <58>
123 DATA227,236,227,234,227,231,227,230                         <150>
124 DATA227,239,223,238,223,236,223,234                         <247>
125 DATA223,231,223,230,223,227,223,239                         <201>
1010 DATA 227,230,230,223,227,223,0,223                         <206>
1011 DATA 227,223,234,223,227,223,5,5                           <66>
1012 DATA 5,5,5,5,5,5,5,5                                       <54>
1013 DATA 5,5,5,6,6,6,6,6                                       <115>
1014 DATA 7,7,8,9,9,10,10,11                                    <32>
1015 DATA 11,11,11,11,10,9,9,9                                  <132>
1016 DATA 9,8,8,7,6,6,6,6                                       <164>
1017 DATA 6,6,5,5,4,4,4,4                                       <13>
1018 DATA 3,3,3,3,2,2,2,2                                       <120>
1019 DATA 2,1,1,1,1,1,1,1                                       <31>
1020 DATA 1,1,1,1,1,1,2,2                                       <60>
1021 DATA 3,3,3,4,4,4,4,5                                       <251>
1022 DATA 5,5,5,6,7,7,7,7                                       <176>
1023 DATA 8,8,8,8,8,8,8,9                                       <41>
1024 DATA 9,9,9,9,9,9,9,9                                       <99>
1025 DATA 10,10,10,10,9,9,9,9                                   <226>
1026 DATA 9,9,9,8,8,7,7,7                                       <255>
1027 DATA 7,7,7,7,7,7,7,7                                       <214>
1028 DATA 7,7,7,7,7,7,7,7                                       <215>
1029 DATA 7,7,7,7,7,7,7,7                                       <216>
1030 DATA 7,7,7,7,7,7,7,7                                       <217>
1031 DATA 6,6                                                   <86>
  ENDE DES LISTINGS
```

Teil 5:

```
0 POKE36869,255:POKE36879,24:PRINT"{CLEA
R}";                                                            <32>
9 PRINT"{GREEN}12"                                              <202>
10 PRINT"3{BLUE}4{SPACE5 GREEN}LM{BLUE                          <117>
11 PRINT"{SPACE}5{SPACE}G{SPACE}G{SPACE}
ND{SPACE2}G{SPACE3}GGG"                                         <170>
12 PRINT"{SPACE2}G{SPACE}G{SPACE}G{SPACE
3}G{SPACE}G{SPACE2}G{SPACE}HG"                                  <221>
13 PRINT"{SPACE2}G{SPACE3}G{SPACE3}GGG{S
PACE2}G{SPACE2}G{SPACE}GG";                                     <252>
14 PRINT"{SPACE}G{SPACE3}H{SPACE}G{SPACE
}G{SPACE3}G{SPACE}G{SPACE2}G"                                   <238>
15 PRINT"{SPACE}G{SPACE5}G{SPACE}G{SPACE
 RED}Y{BLUE SPACE}G{SPACE}GGG"                                  <150>
16 PRINT"{SPACE11 RED}Z[\]^{BLUE                                <119>
17 PRINT"{SPACE12 RED}_!#$%{BLUE                                <1>
18 PRINT"{SPACE5 GREEN}W{BLUE                                   <68>
20 PRINT"{SPACE}GGG{SPACE GREEN}X{BLUE}G
{SPACE3}G{SPACE2}GGG{SPACE2}G{SPACE}G"                          <193>
21 PRINT"{SPACE}G{SPACE2}G{SPACE}G{BLACK
}RS{BLUE SPACE}G{SPACE}G{SPACE5}G{SPACE}
G"                                                              <191>
22 PRINT"{SPACE}GGG{SPACE2}G{BLACK}TU{BL
UE SPACE}G{SPACE}G{SPACE5}G{SPACE}G"                            <138>
23 PRINT"{SPACE}GG{SPACE3}G{SPACE3}G{SPA
CE2}GGG{SPACE2}GGG"                                             <208>
24 PRINT"{SPACE}G{SPACE}G{SPACE2}G{SPACE
3}G{SPACE5}G{SPACE}G{SPACE}G"                                   <190>
25 PRINT"{SPACE}G{SPACE2}G{SPACE}G{SPACE
2}IG{GREEN}%&{BLUE SPACE3}G{SPACE}G{SPAC
E}G"                                                            <158>
26 PRINT"{SPACE}G{SPACE2}G{SPACE2}GGG{SP
ACE GREEN}'({BLUE}GGG{SPACE2}G{SPACE}G"                         <26>
27 PRINT"{GREEN}V{SPACE BLACK}/0{SPACE7
RED})*                                                          <96>
28 PRINT"{SPACE2 BLACK}-.{SPACE7 RED}+,                         <217>
29 PRINT"{SPACE18 GREEN}67                                      <229>
30 PRINT"{SPACE18 GREEN}89                                      <64>
31 PRINT"{SPACE18 GREEN}:;{UP4 WHITE}"                          <199>
32 GOSUB40:POKE46,21:POKE45,92:CLR:LOAD                         <15>
40 FORA=5973TO6129:READB:POKEA,B:C=C+B:N
EXT                                                             <232>
41 IFC<>24128THENPRINT"{CLEAR RVSON BLAC
K}DATA{SPACE}ERROR":END                                         <151>
42 RETURN                                                       <184>
50 DATA217,238,217,236,217,234,217,231                          <169>
51 DATA217,230,217,227,217,223,217,0                            <179>
52 DATA0,223,227,230,0,227,230,231                              <206>
53 DATA0,234,0,0,0,230,234,236                                  <170>
54 DATA0,234,231,230,0,230,227,223                              <221>
55 DATA0,223,0,0,0,223,227,230                                  <91>
56 DATA0,231,0,217,0,227,230,231                                <67>
57 DATA0,234,0,0,0,223,227,230                                  <104>
58 DATA0,231,0,217,0,227,230,231                                <69>
59 DATA0,234,0,223,230,223,227,230                              <5>
60 DATA223,223,227,230,0,227,230,231                            <30>
61 DATA0,234,0,0,0,230,234,236                                  <178>
62 DATA0,234,231,230,0,230,227,223                              <229>
63 DATA0,223,0,0,0,223,227,230                                  <99>
64 DATA0,231,0,0,0,227,230,231                                  <60>
65 DATA0,234,0,0,0,230,234,236                                  <182>
66 DATA0,234,231,230,0,230,227,223                              <233>
67 DATA0,223,0,0,223,227,230,230                                <44>
68 DATA223,227,223,0,223,227,223,234                            <220>
69 DATA223,227,223,0,223                                        <46>
  ENDE DES LISTINGS
```

Hauptprogramm:

```
0 POKE55,133:POKE56,22:Y=36876:POKEY-7,2
55:D=36903:E=40:F=7066:G=32:POKEF,32                            <156>
1 POKED,50:I=7150:H=100:PRINT"{CLEAR}":P
OKEY+3,248:J=5765:K=1:GOSUB100                                  <65>
10 C=130:SYS5842:SYSJ:PRINT"{HOME RVSON
BLACK}"TAB(PEEK(I)+2)"MILE"INT(L/18)                            <254>
11 PRINT"{HOME}"TAB(PEEK(I)+25)"{RVSON}M
AN"K"{LEFT}."                                                   <19>
15 WAIT37137,32,32:SYS5809                                      <235>
```

```
16 PRINT"{HOME}"TAB(PEEK(I)+2)"{SPACE2}/
0{SPACE4 DOWN LEFT8 SPACE2}-.{SPACE3}"      <249>
20 FORW=1TO6:SYSJ:FORA=0TOC:NEXT:C=C-C/H
:IFPEEK(F)=GTHENNEXT                        <215>
21 PRINT"{HOME}"TAB(PEEK(I))A$(RND(1)*30
);:L=L+1                                    <129>
22 IFPEEK(F)=GGOTO20                        <137>
24 IFPEEK(F)=8ORPEEK(F)=9THENPOKEF,32:GO
TO20                                        <94>
25 SYS5793:POKEY,0:POKEY-1,0:POKEY+1,220
:K=K+1                                      <97>
30 PRINT"{HOME DOWN12}"TAB(PEEK(6811)-8)
"{BLACK}P{DOWN LEFT}Q";                     <109>
40 FORA=15TO0STEP-.05:POKEY+2,A:NEXT:POK
EY+1,0:POKEF,32:POKEY+2,15                  <113>
50 IFK<4GOTO10                              <95>
55 PRINT"{HOME RVSON}"TAB(180)"┌──────────
──────┐":PRINTTAB(4)"{RVSON}|{SPACE11}|"   <139>
56 PRINTTAB(4)"{RVSON}|{SPACE11}|"          <103>
57 PRINTTAB(4)"{RVSON}|{SPACE11}|":PRINT
TAB(4)"{RVSON}└──────────┘"                 <242>
60 A$(1)="GAME"                             <250>
61 PRINT"{HOME RVSON}"TAB(204);:GOSUB97:
A$(1)="OVER":PRINT"{SPACE}";:GOSUB97:PRI
NT:PRINT                                    <111>
62 PRINTTAB(6)"{RVSON}";:A$(1)=STR$(INT(
L/18)):GOSUB97                              <116>
63 PRINT"{SPACE}";:A$(1)="MILES":IFINT(L
/18)=1THENA$(1)="MILE"                      <4>
64 GOSUB97                                  <228>
90 WAIT37137,32,32:RUN                      <81>
97 FORA=1TOLEN(A$(1)):POKEY,240:A$=MID$(
A$(1),A,1)                                  <251>
98 PRINTA$;:POKEY,0                         <0>
99 FORB=1TO100:NEXT:NEXT:RETURN             <214>
100 DIMA$(30):A$(0)="{RIGHT BLACK RIGHT2
}I"                                         <13>
101 A$(1)="{BLACK RIGHT7}H"                 <215>
102 A$(2)="{RIGHT3 GREEN}LM{DOWN PURPLE
LEFT2}NO"                                   <56>
103 A$(3)="{RIGHT7 GREEN}LM{DOWN LEFT2 P
URPLE}NO"                                   <89>
104 A$(4)="{RIGHT9 BLACK}RS{DOWN LEFT2}T
U"                                          <98>
105 A$(5)="{RIGHT BLACK}RS{DOWN LEFT2}TU
"                                           <187>
106 A$(6)="{RIGHT GREEN}V"                  <242>
107 A$(7)="{RIGHT6 GREEN RIGHT}W{DOWN LE
FT}X"                                       <116>
108 A$(8)="{RED}Y{DOWN LEFT}Z[\]^{DOWN L
EFT4}_!#$"                                  <40>
109 A$(9)="{RIGHT8 GREEN}%&{DOWN LEFT2}'
({DOWN RED LEFT2})*{DOWN LEFT2}+,"          <189>
110 A$(10)="{BLACK RIGHT3}/0{DOWN LEFT2}
-."                                         <79>
111 A$(11)="{RIGHT4 GREEN RIGHT}12{DOWN
LEFT2}3{PURPLE}4{DOWN LEFT2 RIGHT}5"        <209>
112 A$(12)="{GREEN RIGHT3}67{DOWN LEFT2}
89{DOWN LEFT2}:;"                           <200>
113 A$(13)="{RIGHT2 GREEN}12{DOWN LEFT2}
3{PURPLE}4{DOWN LEFT2 RIGHT}5"              <32>
114 A$(14)="{BLACK RIGHT5}/0{DOWN LEFT2}
-."                                         <196>
115 A$(15)="{BLUE RIGHT4}67{DOWN LEFT2}8
9{DOWN LEFT2}:;"                            <103>
116 A$(16)="{RIGHT7 GREEN}%&{DOWN LEFT2}
'({DOWN BLACK LEFT2})*{DOWN LEFT2}+,":RE
TURN                                        <201>
  ENDE DES LISTINGS
```

Geisterhölle für den VC-20

Ghost-Hell

Ziel des Spieles ist es, in der Geisterhölle 8 Pillen aufzusammeln, was auch den Punktestand erhöht. Dies ist aber gar nicht so einfach, weil 3 Monster ihr Unwesen treiben.

Mit der **„SPACE"-Taste** springt Ihr Männchen hoch über die Monster. Ansonsten kann man Leitern rauf und runter gehen und sich fallen lassen – aber nicht zu tief! Andernfalls ist „GAME OVER".

Wenn man das geschafft hat, erscheint ein Schlüssel. Hat man diesen genommen, geht oben links ein Tor auf. Dort muß man hinausgehen. Das nächste Bild beginnt ...

Das Spiel besteht aus 2 Teilen, die separat abgetippt und gespeichert werden müssen. Im Teil 2 ist eine Routine eingebaut, die die „RUN-STOP" und die „RESTORE"-Taste sperrt. Also: Erst abspeichern und dann spielen!!!!!

Über die Tastatur wird gesteuert:

A	=	rechts
D	=	links
F1	=	Leiter rauf
F7	=	Leiter runter
SPACE	=	Springen

```
0 REM****************          *
  *                                                    <145>
1 REM*GHOST HELL  BY*          *
  *                                                    <138>
2 REM*ROLAND SCHOLZ *          *
  *                                                    <88>
3 REM*MARTIN-LUTHER *          *
  *                                                    <144>
4 REM*STRASSE 24     *         *
  *                                                    <48>
5 REM*5354 WEILERSWIST         *                       <109>
6 REM****************                                  <133>
7 REM                                                  <150>
8 REM                                                  <151>
9 REM                                                  <152>
10 PRINT"{CLEAR}":POKE56,28                            <223>
20 FORT=7168TO7248                                     <207>
30 READB                                               <41>
40 IFB=-1THEN1000                                      <63>
50 POKET,B                                             <253>
60 NEXT                                                <190>
70 DATA126,90,126,60,24,24,36,66                       <168>
80 DATA126,219,255,255,231,219,126,0                   <52>
90 DATA31,127,255,255,170,87,82,138                    <20>
100 DATA255,255,255,255,238,229,180,132                <115>
110 DATA248,254,255,255,238,87,82,138                  <139>
120 DATA129,129,255,129,129,255,129,255                <128>
130 DATA64,160,191,165,65,0,0,0                        <148>
140 DATA0,0,0,0,0,0,0,0                                <99>
150 DATA255,145,145,255,255,143,137,137                <210>
160 DATA0,56,124,124,124,56,0,0,-1                     <206>
1000 PRINT"{HOME RVSON RIGHT6}GHOST{SPAC
E}HELL"                                                <72>
1010 PRINT"{DOWN}SIE{SPACE}MUESSEN{SPACE
}MIT{SPACE}IHREM{SPACE DOWN}MAENNCHEN{SP
ACE}ALLE{SPACE}PILLEN{SPACE DOWN}'FRESSE
N'."                                                   <144>
1020 PRINT"{DOWN}ABER{SPACE}PASSEN{SPACE
}SIE{SPACE}AUF,{SPACE2 DOWN}DIE{SPACE}MO
NSTER{SPACE}WOLLEN{SPACE4 DOWN}SIE{SPACE
}DARAN{SPACE}HINDERN."                                 <164>
1030 PRINT"{DOWN}UEBER{SPACE}DIE{SPACE}M
ONSTER{SPACE5 DOWN}KOENNEN{SPACE}SIE{SPA
CE}SPRINGEN{SPACE2 DOWN}MIT{SPACE}DER{SP
ACE}TASTE{SPACE RVSON}SPACE{RVSOFF}."                  <199>
1040 PRINT"{DOWN2 RIGHT RVSON}TASTE":POK
E198,0:WAIT198,1:FORT=38TO160:POKE36881,
T:NEXT:PRINT"{CLEAR}"                                  <106>
1051 PRINT"NUN{SPACE}ERSCHEINT{SPACE}EIN
{SPACE5 DOWN}SCHLUESSEL,DEN{SPACE}MUESSE
N{DOWN}SIE{SPACE}HOHLEN"                               <62>
1052 PRINT"{DOWN}DANN{SPACE}GEHT{SPACE}N
AEMLICH{SPACE}DAS{DOWN}TOR{SPACE}OBEN{SP
ACE}LINKS{SPACE}AUF{SPACE4 DOWN}DORT{SPA
CE}MUESSEN{SPACE}SIE{SPACE6 DOWN}RAUSGEH
EN"                                                    <222>
1053 PRINT"{DOWN}DANACH{SPACE}FAENGT{SPA
CE}DAS{SPACE5 DOWN}SPIEL{SPACE}VON{SPACE
}FORN{SPACE}AN."                                       <168>
1054 FORT=160TO38STEP-1:POKE36881,T:NEXT               <120>
1055 PRINT"{DOWN2 RIGHT RVSON}TASTE{RVSO
FF}":POKE198,0:WAIT198,1:PRINT"{CLEAR}"                <52>
1060 PRINT"{RIGHT3 SPACE3}STEUERUNG{DOWN
 LEFT9}"                                               <197>
1070 PRINT"LINKS{SPACE}:{SPACE}A"                      <81>
1075 PRINT                                             <208>
1080 PRINT"RECHTS:{SPACE}D"                            <212>
1085 PRINT                                             <218>
1090 PRINT"RAUF{SPACE2}:{SPACE}F1"                     <156>
1095 PRINT                                             <228>
1100 PRINT"RUNTER:{SPACE}F7"                           <2>
1109 PRINT"{DOWN2}SPRINGEN:{SPACE3 RVSON
}SPACE{RVSOFF}"                                        <46>
1110 PRINT"{DOWN2 RIGHT RVSON}TASTE":POK
E198,0:WAIT198,1:FORT=38TO160:POKE36881,
T:NEXT                                                 <178>
1120 PRINT"{CLEAR}LADEN{SPACE}SIE{SPACE}
BITTE{SPACE}DEN{SPACE RIGHT2 DOWN}2.TEIL
{SPACE}EIN{SPACE}!"                                    <172>
1130 FORT=160TO38STEP-1:POKE36881,T:NEXT
:NEW                                                   <6>
  ENDE DES LISTINGS
```

Teil 2.

```
0 POKE37150,2:POKE788,194                              <99>
5 POKE36878,15:D1=1:R1=0:A=0:Q=0:W=0:E=0
:R=0                                                   <123>
10 FORT=7680TO8186:POKET,7:NEXT:POKE3686
9,255:RESTORE                                          <86>
11 FORT=7424TO7424+7:READS:POKET,S:NEXT                <105>
12 DATA0,0,0,0,0,0,0,0                                 <227>
13 POKE36879,14:PRINT"{WHITE}"                         <17>
100 PRINT"{CLEAR}"                                     <130>
110 PRINT                                              <7>
115 PRINT                                              <12>
120 PRINT"{SPACE13}I{SPACE3}E{SPACE}HH"                <33>
130 PRINT"{SPACE}HH{SPACE}BCCCCCCCCCCDE
{SPACE}HH"                                             <42>
140 PRINT"{SPACE}HH{SPACE14}E{SPACE}HH"                <13>
150 PRINT"{SPACE}HH{SPACE14}E{SPACE}HH"                <23>
160 PRINT"{SPACE}HH{SPACE14}E{SPACE}HH"                <33>
170 PRINT"{SPACE}HH{SPACE}I{SPACE9}E{SPA
CE2}E{SPACE}HH"                                        <191>
180 PRINT"BCCCCCCCCCCCDE{SPACE2}E{SPACE
}HH"                                                   <126>
190 PRINT"{SPACE6}HH{SPACE6}E{SPACE2}E{S
PACE}HH"                                               <68>
200 PRINT"{SPACE6}HH{SPACE6}E{SPACE2}E{S
PACE}HH"                                               <78>
210 PRINT"{SPACE}I{SPACE}E{SPACE2}HH{SPA
CE2}I{SPACE3}E{SPACE2}E{SPACE}HH"                      <239>
220 PRINT"{SPACE}BCED{SPACE}HH{SPACE2}BC
CCCCDE{SPACE}HH"                                       <84>
230 PRINT"{SPACE}E{SPACE}E{SPACE2}HH{SPA
CE9}E{SPACE}HH"                                        <105>
240 PRINT"BE{SPACE}E{SPACE2}HH{SPACE}B{S
PACE7}E{SPACE}HH"                                      <113>
250 PRINT"{SPACE}E{SPACE}E{SPACE3}I{SPAC
E6}E{SPACE2}EIHH"                                      <241>
260 PRINT"{SPACE}E{SPACE}E{SPACE3}BCCCCD
{SPACE}EBCCDHH"                                        <20>
270 PRINT"{SPACE}E{SPACE}E{SPACE5}HH{SPA
CE3}E{SPACE4}HH"                                       <19>
280 PRINT"{SPACE}E{SPACE}E{SPACE2}I{SPAC
E2}HH{SPACE3}E{SPACE4}HH"                              <142>
290 PRINT"{SPACE}E{SPACE}BCCCCDHH{SPACE3
}E{SPACE4}HH"                                          <71>
```

```
300 PRINT"{SPACE}E{SPACE7}I{SPACE4}E{SPA
CE4}HH"                                  <87>
310 PRINT"BCCCCCCCCCCCCCCCCCDHH"         <170>
311 B=7734:N=7845:M=8022:A=7728:POKEA+Q+
W+E+R,0                                  <14>
320 GETA$                                <208>
330 IFA$="A"THEN500                      <175>
340 IFA$="D"THEN600                      <212>
350 IFA$="{F1}"THEN12100                 <6>
360 IFA$="{F7}"THEN12000                 <23>
370 IFA$="{SPACE}"THEN11000              <167>
380 POKEB+Z,7:POKEX+N,7:POKEV+M,7        <12>
390 Z=Z+1:X=X+1:V=V+1                    <216>
400 IFZ>2THENZ=-5                        <29>
410 IFX>2THENX=-6                        <32>
430 IFV>0THENV=-4                        <10>
440 IFZ+B=A+Q+W+E+RTHEN10000             <237>
450 IFX+N=A+Q+W+E+RTHEN10000             <35>
470 IFV+M=A+Q+W+E+RTHEN10000             <47>
480 POKEZ+B,1:POKEX+N,1:POKEV+M,1        <86>
490 FORT=0TO100:NEXT:ONDIGOTO320,7000,11
000                                      <26>
500 POKEA+Q+W+E+R,7:Q=Q-1:GOSUB3000      <60>
510 IFS=8ORS=2ORS=4THENQ=Q+1             <157>
520 GETA$:IFA$="{SPACE}"THEN7000         <65>
530 POKEA+Q+W+E+R,0:POKE36875,225:POKE36
875,0                                    <142>
540 GOTO320                              <99>
600 POKEA+Q+W+E+R,7:W=W+1:GOSUB3000      <68>
610 IFS=8ORS=2ORS=4THENW=W-1             <199>
620 GETA$:IFA$="{SPACE}"THEN7000         <165>
630 GOTO530                              <196>
3000 S=PEEK(A+Q+W+E+R)                   <50>
3010 IFS=1THEN10000                      <161>
3020 IFS=9THENGOSUB9000                  <50>
3030 IFA+Q+W+E+R=7834THENGOTO5           <179>
3035 IFS=2ORS=4ORS=8THEN3060             <46>
3040 IFS=6THENGOSUB8000                  <51>
3050 IFS=2ORS=4ORS=8THEN3060             <61>
3060 D=PEEK(A+Q+W+E+R+22)                <52>
3070 IFD=7ORD=32THEN7000                 <129>
3080 RETURN                              <162>
7000 D1=2:POKEA+Q+W+E+R,7                <238>
7010 E=E+22                              <195>
7020 D2=PEEK(A+Q+W+E+R)                  <56>
7030 IFD2=7ORD2=32THEN7040               <82>
7035 D1=1:D3=0:E=E-22:POKEA+Q+W+E+R,0:GO
TO320                                    <2>
7040 POKEA+Q+W+E+R,0:POKE36875,225:POKE3
6875,0                                   <21>
7050 D3=D3+1                             <199>
7060 IFD3>3THEN10000                     <127>
7070 GOTO380                             <16>
8000 SC=SC+100                           <108>
8010 FORT=7747TO7836STEP22:POKET,7:POKE3
6875,240:POKE36875,0:NEXT                <112>
8020 FORT=7748TO7837STEP22:POKET,7:POKE3
6875,200:POKE36875,0:NEXT                <21>
8030 RETURN                              <11>
9000 FORT=200TO240:POKE36875,T:POKE36875
,0:NEXT                                  <170>
9010 SC=SC+10:SP=SP+1:IFSP=8THENPOKE7933
,6:SP=0                                  <83>
9020 PRINT"{RVSON HOME}SCORE:";SC        <116>
9030 RETURN                              <247>
10000 PRINT"{RVSON HOME DOWN6 SPACE7}GAM
E{SPACE}OVER{SPACE5}"                    <171>
10010 PRINT"{RVSON SPACE}PRESS{SPACE}A{S
PACE}KEY{SPACE}TO{SPACE}START{SPACE5}THE
{SPACE}GAME{SPACE}AGAIN{SPACE4}":POKE198
,0:WAIT198,1:RUN                         <59>
11000 D1=3:POKEA+Q+W+E+R,7               <161>
11010 R1=R1+1                            <135>
11020 IFR1>2THEN11050                    <82>
11030 R=R-22:POKEA+Q+W+E+R,0             <233>
11040 GOTO380                            <162>
11050 E=E+22                             <155>
11060 A9=PEEK(A+Q+W+E+R)                 <27>
11070 IFR1=5THENE=E-22:POKEA+Q+W+E+R,0:R
1=0:D1=1:GOTO320                         <157>
11090 POKEA+Q+W+E+R-22,7:POKEA+Q+W+E+R,0 <51>
11100 GOTO380                            <222>
12000 POKEA+Q+W+E+R,5:E=E+22             <27>
12010 A2=PEEK(A+Q+W+E+R-22)              <184>
12020 IFNOTA2=5THENE=E-22                <40>
12030 IFNOTPEEK(A+Q+W+E+R)=5THENE=E-22   <217>
12040 POKEQ+W+E+R+A,0:POKE36875,225:POKE
36875,0                                  <59>
12050 GOTO320                            <134>
12100 POKEA+Q+W+E+R,5:R=R-22             <23>
12110 A2=PEEK(A+Q+W+E+R)                 <43>
12120 IFNOTA2=5THENR=R+22                <108>
12130 POKEA+Q+W+E+R,0:POKE36875,225:POKE
36875,0                                  <11>
12140 GOTO320                            <224>
```

Tips & Tricks für den VC-20

SCROLLING TIPS

Scroll nach unten weg und zurück.

```
10 FOR X=25 TO130:POKE36881,X:NEXT:PRINT"clearscreen"
20 PRINT"neuer Text"
30 FOR X=130TO25STEP-1:POKE36881,X:NEXT
```

Diese Routine existiert in verschiedenen Varianten, die sich lediglich in den Werten für die X-Schleife unterscheiden.

Scroll nach links weg und zurück.

```
10 FORI=0TO30:POKE251,I:POKE252,16:FOR K=1TO130:NEXT
K,I:PRINT"clearscreen"
20 PRINT"neuer Text"
30 FORI=30TO1STEP-1:POKE251,I:POKE252,16:
   FOR K=1TO10:NEXT K?I
```

K.O.-Scroll.

Folgende Scrollroutine eignet sich besonders für Spiele, in denen die Spielfigur K.O.geschlagen wird – wie z. B. bei 'Kong'-Spielen:

```
10 FOR I=5TO55:POKE 36864,I:POKE36865,I+I:FOR K=1TO65:NEXT
K,I:PRINT"clearscreen"
20 PRINT"neuer Text"
30 POKE36864,12:POKE36865,38
```

Aus- und einblenden.

```
10 FOR I=23TO1 STEP-1:POKE36867,I+I:POKE36866,I-1:POKE36864,13:POKE36865,82-INT(I/2)*4:NEXT I
20 PRINT"clearscreen":PRINT"neuer Text"
30 FOR I=1TO23:POKE36867,I+I:POKE36866,I-1:POKE36864,13:POKE36865,82-
30 FOR I=1TO23:POKE36867,I+I:POKE36866,I-1:POKE36864,13:POKE36865,82-INT(I/2)*4:NEXT I:I=0
```

Seitwärts mit Text

```
10 BL$="19 leertasten": A$="text"
20 A$=BL$+A$
30 FOR I=1 TO LEN(A$): PRINT TAB(2)
   MID$(A$,I,18)
40 PRINT "up up": FOR K=1TO150:
   NEXT K,I (ggfls. :RETURN)
```

EINE Bildschirmzeile löschen.

POKE781,Zeilennummer:POKE782,0:POKE783,0:SYS60045

EINE Bildschirmzeile bis zur angegebenen Spalte löschen:

POKE781,Zeilennummer:POKE782,Spaltennummer:POKE783,0:SYS60047

Erdbeben

```
10 FOR Y=1TO20:FOR X=3 TO 7:POKE 36864,X:NEXT X,Y:POKE 36864,12
20 FOR Y=1TO20:FOR X=23TO27:POKE 36865,X:NEXT X,Y:POKE 36865,38
```

Die erste Zeile ist für ein horizontales, die zweite für ein vertikales Beben.

Scroll nach unten: Variante

```
10 A$="":FOR T= 0 TO 18:A$=
   A$+CHR$(29):NEXT
11 A$=CHR$(19)+A$+CHR$(160)+CHR$(157)+CHR$(157)
12 A$=A$+CHR$(157)+CHR$(157)+CHR$(157)+CHR$(148)+CHR$(148)+CHR$(148)
```

Damit sind die einzusetzenden Strings definiert: der Aufruf erfolgt mit GOTO oder GOSUB folgende Zeile:

```
nnnnn PRINT A$:POKE217,158:POKE218,158
```

Schwarzes Loch

```
10 A=150:B=174
20 FOR C=1TO21:A=A-1:B=B-2:POKE 36866,A:POKE 36867,B:NEXT
30 PRINT CHR$(147);
40 FOR C=1TO21:A=A+1:B=B+2:POKE 36866,A:POKE 36867,B:NEXT
(50 RETURN) falls erforderlich
```

Wirbelsturm

```
10 A=36864:B=36865
20 FOR X=46 TO 0 STEP -2:POKEB,X:POKEA,51-X
30 FOR T=1TO5: NEXT T,X
40 ggfl. neues Bild erzeugen, sonst weglassen
50 FOR X= 0to 46Step 2:POKEB,X:POKEA,51-X
60 FOR T=1TO5: NEXT T,x
```

Zeilen 20 und 50 können als selbständige Unterroutinen aufgerufen werden, wenn die Zeile mit einem RETURN beendet wird.

Wieder auswärts

```
10 PRINT"clear 19(neunzehnmal) down"
20 FOR N=1TO LEN(A$):REM A$ ist der zu scrollende Text
30 PRINTTAB(11)MID$(A$,N,1)
40 FOR T=1TO175:NEXT T,N
50 FOR K=1TO500:NEXT:PRINTCHR$(147) (ggfls. :RETURN)
```

Langsam aufwärts

```
10 POKE 36879,8:PRINT" text"
20 FOR A=131TO0 STEP-1: POKE 36865,A: FOR B=1 TO45:
   NEXT B,A: POKE 36865,24
```

Die Aufwärtsgeschwindigkeit wird durch die B Schleife kontrolliert; die durch diese Routine geänderten Farbwerte müssen ggfl. zurückgesetzt werden.

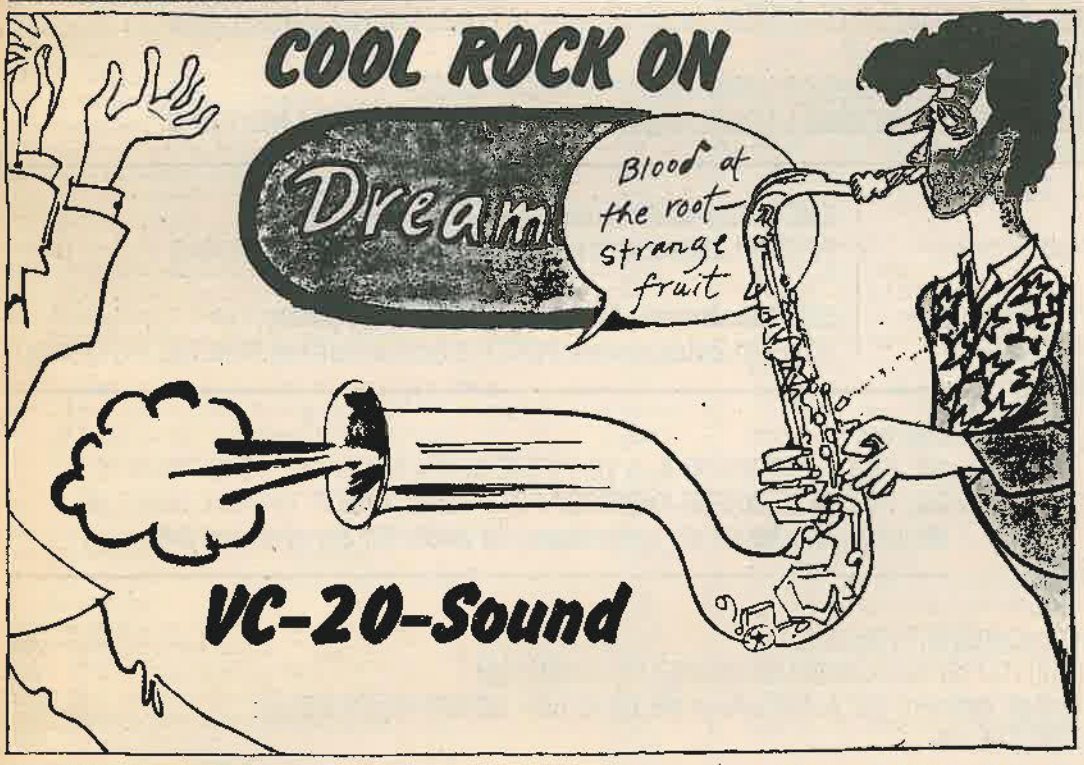
Scroll für eine Zeile nach unten (erste Zeile ausgenommen).

Diese Routine scrollt ab der zweiten Bildschirmzeile eine Zeile nach unten:

```
PRINT"home crsrdown crsrleft inst":POKE218,158
```

In Verbindung mit einer Zählschleife, z. B. FOR I=1TO20, kann man den ganzen Bildschirm nach untern wegscrollen, und die erste Zeile als Statuszeile stehenlassen.

Werkstatt:



Dies ist eine außerordentlich kurze Soundroutine für den VC-20, die voll in Basic geschrieben ist und alle 4 Soundregister ausnutzt.

Mit diesem kleinen Programm läßt sich sehr schön experimentieren. Z.B. können die Werte in Zeile 120 geändert werden.

Versuchen Sie mal
250 AB$="22324322"

Wir fanden, daß das Programm die Soundmöglichkeiten des VC-20 wirklich optimal ausnutzt.

```
160 POKE36879,253:REM COOL ROCK                          <72>
170 PRINT"{CLEAR}"                                       <200>
180 PRINT"{RIGHT6}SILVER-SOFT"                           <62>
191 PRINT"{DOWN RIGHT8 BLACK}P{RED}R{CYA
N}E{PURPLE}S{GREEN}E{BLUE}N{YELLOW}T{BLA
CK}:"                                                    <142>
200 PRINTCHR$(144):PRINT"(C)'86{SPACE}BY
{SPACE}THOMAS{SPACE}WELP"                                <94>
201 PRINT"{RIGHT2 DOWN RIGHT4}COOL{SPACE
}ROCK"                                                   <124>
210 B=36874:N=36875:M=36876:J=36877                      <207>
220 R(1,1)=195:R(1,2)=207:R(1,3)=215:R(1
,4)=219:R(1,5)=221:R(1,6)=219:                           <186>
221 R(1,7)=215:R(1,8)=207                                <90>
230 R(2,1)=209:R(2,2)=219:R(2,3)=225:R(2
,4)=228:R(2,5)=229:R(2,6)=228:                           <101>
231 R(2,7)=225:R(2,8)=219                                <188>
240 R(3,1)=215:R(3,2)=223:R(3,3)=228:R(3
,4)=231:R(3,5)=232:R(3,6)=231:                           <46>
241 R(3,7)=228:R(3,8)=223                                <139>
250 AB$="11213211"                                       <44>
260 TA$="101202"                                         <23>
270 T(1,1)=207:T(2,1)=215:T(3,1)=225                     <49>
280 T(1,2)=207:T(2,2)=215:T(3,2)=225                     <107>
290 T(1,3)=209:T(2,3)=219:T(3,3)=225                     <13>
300 T(1,4)=207:T(2,4)=215:T(3,4)=225                     <223>
310 T(1,5)=215:T(2,5)=223:T(3,5)=228                     <88>
320 T(1,6)=209:T(2,6)=219:T(3,6)=225                     <187>
330 T(1,7)=207:T(2,7)=215:T(3,7)=225                     <141>
340 T(1,8)=207:T(2,8)=215:T(3,8)=225                     <199>
350 POKE36878,5                                          <29>
360 FORU=1TO6                                            <230>
370 FORZ=1TO8                                            <6>
380 RESTORE                                              <9>
390 AB=0:ZE=2:ZT=0                                       <81>
400 FORX=0TO23                                           <93>
410 READC:POKEM,T(C,Z)                                   <164>
420 ZE=ZE+1:IFZE=3THENZE=0:AB=AB+1                       <109>
430 POKEN,R(VAL(MID$(AB$,Z,1)),AB)                       <163>
440 ZT=ZT+1:IFZT=7THENZT=1                               <18>
450 A=VAL(MID$(TA$,ZT,1)):IFA=1THENPOKEJ
,200                                                     <35>
460 IFA=2THENPOKEJ,252                                   <229>
470 IFX=0ORX=11THENPOKEB,R(VAL(MID$(AB$,
Z,1)),1)                                                 <150>
480 IFX=5ORX=17THENPOKEB,R(VAL(MID$(AB$,
Z,1)),3)                                                 <38>
490 POKEJ,0                                              <90>
500 FORW=1TO15:NEXTW,X,Z                                 <191>
510 ONUGOSUB550,610,550,610,550,610                      <47>
520 NEXTU:PRINT"{CLEAR}":END                             <162>
530 DATA1,1,2,3,3,1,2,2,3,1,1,2,3,3,1,2,
2,3,3,3                                                  <165>
540 DATA3,3,3,3                                          <49>
550 PRINT"{CLEAR DOWN2}DIESES{SPACE}MUSI
C-PROGRAMM{DOWN3 RIGHT2}***IST{SPACE}COO
L{SPACE}ROCK***"                                         <23>
560 PRINT                                                <203>
570 PRINT                                                <213>
580 PRINT                                                <223>
590 PRINT                                                <233>
600 RETURN                                               <232>
610 PRINT"{CLEAR DOWN4}";CHR$(144)                       <64>
620 PRINT"DIE{SPACE}MELODIE{SPACE}BEINHA
LTET{DOWN2}4{SPACE}STIMMEN"                              <6>
630 PRINT"{DOWN}1)3-TON{SPACE}MELODIE"                   <184>
631 PRINT"{DOWN}2)JAZZ-UNTERSTIMME"                      <186>
632 PRINT"{DOWN}3)WECHSELBASS"                           <172>
640 PRINT"{DOWN}4)SCHLAGZEUG":PRINTCHR$(
144)                                                     <163>
650 PRINT                                                <37>
660 RETURN                                               <36>
```

ROADRUNNER

3 Spiele in einem Programm für den C-64

„Roadrunner" ist ein Superspiel, ganz in Maschinensprache geschrieben und beinhaltet 3 verschiedene Spielfelder.

Level 1:

Im ersten Bild muß der Roadrunner von Sonnenaufgang bis Sonnenuntergang durch die Wüste laufen und dabe viele Hindernisse überwinden.

Joysticksteuerung:
nach oben = Sprung
nach oben und dann nach rechts verlängert den Sprung

Level 2:

Der Roadrunner hat sich aus der Wüste in die Berge gerettet. Nun muß er über Schluchten springen.

Joysticksteuerung:
Feuerknopf: Roadrunner beginnt zu laufen
nach oben = Großer Sprung
nach rechts= kleiner Sprung
Sind alle 3 Level geschafft, geht das Spiel mit erhöhter Geschwindigkeit von vorne los.

Level 3:

Roadrunner bleibt vom Pech verfolgt. Nun hat er sich in ein Indianerreservat verirrt und muß den Pfeilen der Indianer ausweichen.

Joysticksteuerung:
nach allen 4 Seiten

```
1000 REM DATA-LOADER FUER * ROADRUNNER *   <3>
1010 REM VOR DEM STARTEN ABSPEICHERN       <216>
1020 :                                     <57>
1030 :                                     <68>
1040 DATA0,55,8,194,7,158,53,49,50,49,58
,143,32,42,32,82,79,65,68,82,85,78,78      <39>
1050 DATA69,82,32,42,32,32,32,32,32,32,3
2,32,32,32,32,32,32,32,40,67,41,32,82      <41>
1060 DATA46,32,68,73,83,83,69,82,0,0,0,6
8,68,68,68,68,68,68,68,68,68,40,67         <90>
1070 DATA41,32,82,46,32,68,73,83,83,69,8
2,0,0,0,133,250,169,54,133,251,169,0       <94>
1080 DATA133,252,169,34,133,253,169,1,13
3,98,169,54,133,99,173,161,31,24,105       <9>
1090 DATA10,141,161,31,201,10,144,3,76,1
25,8,238,160,31,173,163,31,0,0,0,0,0       <234>
1100 DATA0,0,0,0,0,0,80,0,1,180,0,1,185,
0,1,164,0,1,144,0,0,64,0,0,64,0,0,64       <210>
1110 DATA0,0,64,0,0,64,64,0,64,80,1,64,2
1,86,144,26,170,164,26,191,164,26,250      <26>
1120 DATA164,26,170,144,6,170,64,0,7,255
,0,1,220,0,1,220,0,1,220,0,1,206,0,1       <177>
1130 DATA134,0,1,131,0,3,3,0,3,1,128,6,0
,192,12,0,104,14,0,112,28,0,88,4,0,0       <171>
1140 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,7,255,0,0,48,0,0,48,0,0       <136>
1150 DATA48,0,0,48,0,0,56,0,0,56,0,0,48,
0,0,48,0,0,48,0,0,48,0,0,56,0,0,124        <247>
```

```
1160 DATA0,0,90,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0      <189>
1170 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,80,0,1,180,0,1,185,0,1,164,0,1,144           <20>
1180 DATA0,0,64,64,0,64,80,1,64,21,86,14
4,26,170,164,26,191,164,26,250,164,26          <91>
1190 DATA170,144,6,170,64,0,0,56,0,0,56,
0,0,56,0,0,124,0,0,254,0,1,255,0,3,255         <198>
1200 DATA128,7,255,192,7,255,192,7,255,1
92,7,255,192,7,255,192,3,255,128,1,255         <181>
1210 DATA0,0,254,0,0,254,0,0,124,0,0,56,
0,0,56,0,0,124,0,0,84,0,0,0,59,128,0           <39>
1220 DATA57,0,0,57,0,0,125,0,0,255,0,1,2
39,0,3,215,128,3,187,192,7,125,192,6           <160>
1230 DATA254,192,7,255,192,7,255,128,3,2
55,128,1,255,0,1,254,0,1,254,0,1,124           <218>
1240 DATA0,1,56,0,3,184,0,0,124,0,0,84,0
,0,3,184,0,1,56,0,1,56,0,1,124,0,1,254         <159>
1250 DATA0,1,255,0,3,255,128,7,255,128,7
,255,192,7,239,192,7,215,192,3,187,192         <103>
1260 DATA3,125,128,1,255,0,0,255,0,0,255
,0,0,125,0,0,57,0,0,59,128,0,124,0,0           <36>
1270 DATA84,0,0,0,0,0,0,0,0,0,0,0,0,32,0
,0,112,0,0,168,0,0,32,0,0,32,0,0,32            <62>
1280 DATA0,0,32,0,0,32,0,0,32,0,0,32,0,0
,32,0,0,32,0,0,32,0,0,32,0,0                   <108>
1290 DATA32,0,0,0,0,0,0,0,0,0,48,0,0,120
,0,0,120,0,24,120,0,31,248,96,7,248            <163>
1300 DATA224,0,255,224,0,127,192,0,124,0
,60,120,0,61,126,0,15,254,0,15,248,0           <215>
1310 DATA0,120,224,0,127,224,0,127,192,0
,124,0,0,120,0,0,120,0,0,252,0,1,254           <15>
1320 DATA0,0,0,0,0,0,1,224,0,7,240,0,14,
248,0,29,120,0,58,208,0,117,176,0,235          <102>
1330 DATA96,1,214,192,3,173,128,7,91,0,1
5,182,0,29,236,0,24,216,0,24,240,0,13          <190>
1340 DATA224,0,7,128,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,240,0,3,248,0,0,28          <32>
1350 DATA0,0,14,0,0,14,0,6,111,0,12,207,
15,12,207,63,217,143,59,249,191,31,255         <244>
1360 DATA255,15,255,255,3,255,254,1,255,
252,0,255,248,1,143,240,1,140,96,3,24          <35>
1370 DATA96,6,24,192,12,48,192,0,0,0,0,0
,0,0,1,255,192,3,255,240,255,255,255           <150>
1380 DATA127,255,254,31,255,252,31,255,2
48,31,255,248,63,255,252,63,255,248            <140>
1390 DATA31,255,252,3,255,248,3,255,240,
7,255,224,7,255,240,3,255,240,7,255            <96>
1400 DATA248,15,255,248,31,255,240,15,25
5,224,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0            <208>
1410 DATA0,7,0,0,61,248,0,15,254,0,63,14
,0,0,14,0,7,252,0,15,248,0,12,0,0,7            <22>
1420 DATA255,224,15,255,241,12,0,50,15,2
55,226,7,255,242,12,0,50,15,255,254            <75>
1430 DATA7,255,252,0,0,48,12,24,17,240,4
,34,16,4,50,8,4,18,8,252,10,8,11,140           <46>
1440 DATA8,137,72,240,135,81,240,192,83,
8,112,110,42,24,60,110,8,16,204,12,48          <33>
1450 DATA152,7,176,176,1,183,224,0,252,0
,0,120,0,0,56,0,0,56,0,0,254,0,0,0,0           <230>
1460 DATA0,0,0,0,0,0,0,4,65,17,85,85,85,
85,85,85,90,85,85,85,149,165,85,150            <14>
1470 DATA85,85,86,85,89,85,85,89,85,85,9
0,85,85,85,85,101,85,85,149,85,86,85           <157>
1480 DATA85,85,85,86,85,85,101,85,85,101
,85,85,85,85,85,0,85,101,85,85,101,85          <211>
1490 DATA149,149,85,149,85,101,85,85,101
,85,85,149,85,86,85,89,89,85,101,101           <254>
1500 DATA85,101,101,85,85,101,85,85,85,8
5,86,85,85,86,85,85,86,86,85,86,86,85          <115>
1510 DATA86,85,165,86,85,89,86,85,89,89,
85,153,101,85,89,0,255,255,255,255,255         <74>
1520 DATA255,255,255,255,255,255,255,255
,251,255,255,238,255,255,255,255,255            <33>
1530 DATA255,255,255,255,255,255,255,255
,255,255,255,255,255,255,238,255,255,251        <179>
1540 DATA255,255,255,255,255,255,255,255
,254,255,255,255,255,251,255,255,255            <144>
1550 DATA255,251,255,255,255,255,255,0,2
55,255,255,255,255,255,255,255,255,255,255      <186>
1560 DATA255,255,255,191,255,255,239,255
,239,255,255,255,255,255,255,255,251            <10>
1570 DATA255,255,254,255,191,255,254,255
,255,255,255,255,255,255,255,255,254            <48>
1580 DATA255,255,255,191,255,191,255,255
,255,255,255,255,255,255,255,255,255            <144>
1590 DATA255,255,0,240,0,255,255,3,255,2
55,255,254,255,251,255,255,255,255,255          <88>
1600 DATA255,255,255,255,255,255,255,255
,254,255,255,255,255,255,255,255,255            <172>
1610 DATA255,255,255,255,255,255,255,255
,191,255,255,255,255,255,255,255,255            <180>
1620 DATA255,255,255,255,255,255,255,255
,255,255,255,239,254,0,170,170,170,170          <142>
1630 DATA170,170,169,170,170,167,106,170
,149,106,170,165,106,170,169,106,170            <49>
1640 DATA169,170,166,169,170,150,169,85,
86,169,85,86,165,170,90,165,101,90,165          <202>
1650 DATA90,106,169,85,106,170,190,170,1
70,186,170,170,186,170,170,186,170,170          <250>
1660 DATA186,170,170,250,170,0,170,170,1
70,170,170,170,170,170,170,149,235,94           <209>
1670 DATA149,105,86,154,105,166,154,105,
166,154,105,166,149,105,166,149,233             <99>
1680 DATA166,150,169,166,151,169,166,153
,169,166,153,233,166,154,105,166,154            <146>
1690 DATA105,86,154,107,94,170,170,170,1
70,170,170,170,170,170,170,170,170,170,0        <181>
1700 DATA170,170,170,170,170,170,170,170
,170,181,233,94,149,105,94,154,105,102          <121>
1710 DATA154,105,166,154,105,166,154,105
,166,149,105,166,149,105,166,154,105            <175>
1720 DATA166,154,105,166,154,105,166,154
,105,102,154,105,94,154,105,94,170,170          <43>
1730 DATA170,170,170,170,170,170,170,170
,170,170,0,170,170,170,170,170,170,170           <102>
1740 DATA170,170,149,233,166,149,105,166
,154,105,166,154,105,166,154,105,166             <207>
1750 DATA149,105,166,149,233,166,150,169
,166,151,169,166,153,169,166,153,233             <23>
1760 DATA166,154,105,166,154,105,86,154,
107,94,170,170,170,170,170,170,170,170           <199>
1770 DATA170,170,170,170,0,170,170,170,1
70,170,170,170,170,170,170,150,105,102,150       <106>
1780 DATA105,102,150,105,102,150,105,102
,150,105,102,150,105,102,149,105,86              <215>
1790 DATA153,105,150,153,105,150,153,105
,150,153,105,150,153,105,150,153,105             <218>
1800 DATA150,153,105,150,170,170,170,170
,170,170,170,170,170,170,170,170,0,170            <242>
1810 DATA170,170,170,170,170,170,170,170
,181,233,94,149,105,86,154,169,166,154           <109>
1820 DATA169,166,154,169,166,149,105,86,
149,105,94,154,169,106,154,169,122,154           <183>
1830 DATA169,154,154,169,158,154,169,166
,149,105,166,181,233,166,170,170,170             <12>
1840 DATA170,170,170,170,170,170,170,170
,170,0,170,170,170,170,170,170,170,170            <237>
1850 DATA170,170,170,170,170,170,170,170
,170,170,170,170,170,170,170,170,170              <59>
1860 DATA170,170,170,170,170,170,170,170
,170,170,170,170,170,170,170,170,170              <69>
1870 DATA170,170,170,170,170,170,170,170
,170,170,170,170,170,170,170,170,170              <79>
```

```
1880 DATA170,170,170,170,0,0,255,0,3,255
,192,15,255,240,31,255,248,63,255,252      <185>
1890 DATA127,255,254,127,255,254,255,255
,255,255,255,255,255,255,255,255,255,255    <177>
1900 DATA255,255,255,255,127,255,254,127
,255,254,63,255,252,15,255,240,3,255        <9>
1910 DATA192,0,255,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,40,0,2,170       <233>
1920 DATA0,10,170,128,10,170,168,2,170,1
68,10,170,170,42,170,170,170,170,170        <80>
1930 DATA170,170,168,42,170,160,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0          <8>
1940 DATA0,0,0,0,0,0,0,0,0,0,32,0,0,32,0
,0,32,0,0,32,0,0,32,0,0,32,0,0,32,0         <217>
1950 DATA0,32,0,0,32,0,0,32,0,0,32,0,0,3
2,0,0,32,0,0,168,0,0,112,0,0,32,0,0         <30>
1960 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0    <214>
1970 DATA0,0,0,0,0,0,16,0,0,56,0,0,124,0
,0,254,0,0,186,0,0,254,0,0,124,0,0,56       <233>
1980 DATA0,0,56,0,0,56,0,0,0,56,0,0,56,0
,0,56,0,0,124,0,0,254,0,1,255,0,3,255       <205>
1990 DATA128,7,255,192,7,255,192,7,255,1
92,7,255,192,7,255,192,3,255,128,1,255      <206>
2000 DATA0,0,254,0,0,254,0,0,124,0,0,56,
0,0,56,0,0,124,0,0,84,0,0,0,59,128,0        <64>
2010 DATA57,0,0,57,0,0,125,0,0,255,0,1,2
39,0,3,215,128,3,187,192,7,125,192,6        <185>
2020 DATA254,192,7,255,192,7,255,128,3,2
55,128,1,255,0,1,254,0,1,254,0,1,124        <243>
2030 DATA0,1,56,0,3,184,0,0,124,0,0,84,0
,0,3,184,0,1,56,0,1,56,0,1,124,0,1,254      <184>
2040 DATA0,1,255,0,3,255,128,7,255,128,7
,255,192,7,239,192,7,215,192,3,187,192      <128>
2050 DATA3,125,128,1,255,0,0,255,0,0,255
,0,0,125,0,0,57,0,0,59,128,0,124,0,0        <62>
2060 DATA84,0,0,170,170,170,170,170,186,
170,170,170,186,170,170,170,171,170         <43>
2070 DATA170,170,170,170,170,170,170,170
,190,182,150,190,181,90,90,169,105,106      <241>
2080 DATA170,85,170,170,150,170,170,149,
170,150,86,106,165,101,170,170,105,106      <0>
2090 DATA169,106,90,173,170,154,175,170,
186,170,170,170,0,0,0,0,0,0,0,0,0,0         <156>
2100 DATA0,32,0,0,154,32,2,85,152,9,117,
88,9,85,88,9,93,224,2,149,88,0,39,86        <25>
2110 DATA0,41,118,0,41,88,0,10,160,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0    <218>
2120 DATA0,170,170,170,170,170,170,171,1
70,174,170,170,170,170,170,170,170,170      <148>
2130 DATA234,170,170,170,171,170,170,170
,170,170,170,170,170,170,170,170,170,170    <110>
2140 DATA170,171,234,170,170,170,170,170
,170,170,170,170,170,170,170,234,170        <182>
2150 DATA170,170,170,186,170,170,170,170
,174,170,170,170,0,0,0,0,0,0,0,0,0,0        <152>
2160 DATA0,32,0,0,112,0,0,168,0,0,32,0,0
,32,0,0,32,0,0,32,0,0,32,0,0,32,0,0         <200>
2170 DATA32,0,0,32,0,0,32,0,0,32,0,0,32,
0,0,32,0,0,32,0,0,0,0,0,0,0,0,0,0,0         <236>
2180 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0    <179>
2190 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,169,4,141,152      <196>
2200 DATA31,32,156,18,238,153,31,173,153
,31,141,251,7,169,25,141,6,208,169,70       <204>
2210 DATA141,7,208,230,252,169,8,141,152
,31,32,156,18,238,153,31,173,153,31         <91>
2220 DATA141,252,7,169,40,141,8,208,169,
135,141,9,208,230,252,169,16,141,152        <225>
2230 DATA31,32,156,18,238,153,31,173,153
,31,141,253,7,169,120,141,10,208,169        <178>
2240 DATA135,141,11,208,230,252,169,32,1
41,152,31,32,156,18,238,153,31,173,153      <141>
2250 DATA31,141,254,7,169,200,141,12,208
,169,135,141,13,208,230,252,169,64,141      <230>
2260 DATA152,31,32,156,18,238,153,31,173
,153,31,141,255,7,169,25,141,14,208         <49>
2270 DATA169,135,141,15,208,230,252,169,
128,141,152,31,32,156,18,76,241,18,165      <206>
2280 DATA250,24,105,4,133,250,165,250,20
1,3,144,3,76,174,18,230,251,160,0,177       <104>
2290 DATA250,200,201,128,240,7,177,250,1
57,39,208,232,96,177,250,200,157,39         <181>
2300 DATA208,232,177,250,200,141,37,208,
177,250,141,38,208,76,221,18,173,28         <111>
2310 DATA208,13,152,31,141,28,208,96,173
,37,208,201,0,240,3,76,211,18,173,38        <3>
2320 DATA208,201,0,240,238,76,211,18,169
,1,141,45,4,169,2,141,55,4,169,3,141        <147>
2330 DATA65,4,169,4,141,75,4,169,5,141,1
09,5,169,6,141,119,5,169,7,141,129,5        <153>
2340 DATA169,8,141,139,5,169,7,141,45,21
6,141,55,216,141,65,216,141,75,216,141      <137>
2350 DATA109,217,141,119,217,141,129,217
,141,139,217,162,16,160,0,24,32,240         <119>
2360 DATA255,169,195,32,210,255,200,192,
40,208,248,162,17,160,15,24,32,240,255      <3>
2370 DATA162,0,189,93,19,32,210,255,232,
224,8,208,245,76,101,19,84,65,70,69         <147>
2380 DATA76,32,32,32,162,19,160,5,24,32,
240,255,162,0,189,125,19,32,210,255         <93>
2390 DATA232,224,29,208,245,76,154,19,95
,32,61,32,69,88,73,84,32,32,32,76,69        <24>
2400 DATA69,82,84,65,83,84,69,32,61,32,8
3,85,67,72,69,78,162,21,160,10,24,32        <5>
2410 DATA240,255,162,0,173,154,31,201,5,
240,33,189,185,19,32,210,255,232,224        <204>
2420 DATA19,208,238,76,204,19,65,45,72,3
2,61,32,83,69,76,69,67,84,32,83,80,82       <94>
2430 DATA73,84,69,173,140,31,201,58,240,
9,141,189,6,238,140,31,76,233,19,169        <59>
2440 DATA49,141,189,6,169,48,141,190,6,2
38,140,31,96,162,23,160,11,24,32,240        <77>
2450 DATA255,162,0,173,154,0,234,234,234
,234,234,234,234,234,169,0,32,8,31,234      <25>
2460 DATA234,234,234,234,234,234,169,5,1
41,24,212,169,0,141,4,212,141,11,212        <72>
2470 DATA141,18,212,169,0,141,224,207,14
1,232,3,141,233,3,141,234,3,141,235         <26>
2480 DATA3,76,214,25,169,0,160,0,72,138,
72,152,72,160,0,162,0,200,192,255,144       <82>
2490 DATA251,232,236,249,3,144,245,104,1
68,104,170,104,96,72,138,72,152,72,160      <76>
2500 DATA0,162,0,200,192,245,144,251,232
,236,250,3,144,245,104,168,104,170,104      <118>
2510 DATA96,169,0,160,32,133,250,132,251
,169,0,162,0,160,0,145,250,200,208,251      <86>
2520 DATA230,251,232,224,32,208,242,96,1
69,59,141,17,208,169,24,141,24,208,169      <52>
2530 DATA216,141,22,208,169,0,96,169,27,
141,17,208,169,21,141,24,208,169,200        <25>
2540 DATA141,22,208,96,141,243,207,72,13
8,72,152,72,169,0,133,250,140,234,207       <198>
2550 DATA169,32,133,251,142,233,207,192,
0,240,17,136,230,251,165,250,24,105         <96>
2560 DATA64,133,250,144,240,230,251,76,1
86,20,224,0,240,15,202,165,250,24,105       <82>
2570 DATA8,133,250,144,2,230,251,76,207,
20,234,165,250,24,105,3,133,250,144         <240>
2580 DATA2,230,251,169,5,141,235,207,169
,0,133,252,133,253,174,243,207,24,105       <211>
2590 DATA64,133,252,176,8,202,224,0,240,
8,76,252,20,230,253,76,3,21,162,0,160       <39>
```

```
2600 DATA0,140,246,207,140,245,207,140,2
44,207,173,246,207,201,3,240,37,140      <206>
2610 DATA239,207,142,238,207,172,238,207
,177,252,172,239,207,145,250,238,246     <3>
2620 DATA207,232,165,250,24,105,8,133,25
0,176,3,76,29,21,230,251,76,29,21,234    <188>
2630 DATA169,0,141,246,207,238,245,207,1
73,244,207,201,2,240,62,173,245,207      <41>
2640 DATA205,235,207,240,21,165,250,56,2
33,23,201,233,176,5,133,250,76,29,21     <172>
2650 DATA198,251,133,250,76,29,21,238,24
4,207,169,8,141,235,207,169,0,141,245    <176>
2660 DATA207,230,251,165,250,24,105,33,1
33,250,176,3,76,29,21,230,251,76,29      <72>
2670 DATA21,96,173,245,207,201,8,240,3,7
6,97,21,32,18,22,169,0,133,250,169,4     <131>
2680 DATA133,251,173,242,207,141,225,207
,32,204,21,173,240,207,141,225,207,169   <231>
2690 DATA0,133,250,169,216,133,251,32,20
4,21,104,168,104,170,104,96,172,234      <10>
2700 DATA207,192,0,240,15,136,165,250,24
,105,40,133,250,144,2,230,251,76,207     <31>
2710 DATA21,172,233,207,192,0,240,10,136
,230,250,208,247,230,251,76,229,21,160   <3>
2720 DATA0,173,225,207,145,250,200,192,3
,208,249,160,40,145,250,200,192,43,208   <9>
2730 DATA249,160,80,145,250,200,192,83,2
08,249,96,96,173,17,208,41,16,240,9      <48>
2740 DATA173,17,208,41,239,141,17,208,96
,173,17,208,9,16,141,17,208,96,169,0     <154>
2750 DATA168,153,0,208,200,192,17,208,24
8,96,234,169,0,141,32,208,141,33,208     <247>
2760 DATA32,19,22,32,103,20,32,130,20,32
,19,22,169,14,141,33,208,169,0,141,255   <57>
2770 DATA207,169,37,141,254,207,169,50,1
41,253,207,160,160,140,242,207,160,2     <73>
2780 DATA140,240,207,162,0,160,21,169,59
,32,164,20,232,224,38,208,248,162,0      <110>
2790 DATA200,32,164,20,232,224,38,208,24
8,162,0,160,22,169,54,32,164,20,162      <215>
2800 DATA3,169,55,32,164,20,162,6,169,56
,32,164,20,162,9,169,57,32,164,20,162    <207>
2810 DATA12,169,58,32,164,20,160,240,140
,242,207,160,14,140,240,207,173,224      <23>
2820 DATA207,160,22,162,25,169,53,32,164
,20,162,28,32,164,20,162,31,32,164,20    <221>
2830 DATA162,34,32,164,20,162,37,32,164,
20,173,207,207,201,1,240,1,96,32,228     <130>
2840 DATA22,32,19,22,76,99,24,160,8,140,
242,207,160,7,140,240,207,174,255,207    <193>
2850 DATA160,19,173,253,207,32,164,20,23
2,232,232,224,37,144,243,174,254,207     <189>
2860 DATA32,164,20,174,255,207,160,17,17
3,253,207,24,105,1,32,164,20,232,232     <173>
2870 DATA224,37,144,241,174,254,207,32,1
64,20,174,255,207,160,15,173,253,207     <29>
2880 DATA24,105,2,32,164,20,232,232,232,
224,37,144,240,174,254,207,32,164,20     <119>
2890 DATA162,37,160,16,169,51,32,164,20,
162,5,160,16,169,51,32,164,20,162,12     <223>
2900 DATA160,19,169,51,32,164,20,162,12,
160,17,169,50,32,164,20,96,169,192,133   <81>
2910 DATA250,141,249,207,141,219,207,169
,50,133,251,141,248,207,141,218,207      <87>
2920 DATA169,7,141,252,207,141,247,207,1
69,0,141,246,207,96,162,0,160,0,177      <135>
2930 DATA250,157,60,3,232,200,192,8,208,
245,206,252,207,240,16,230,251,165,250   <224>
2940 DATA24,105,64,133,250,144,2,230,251
,76,132,23,173,249,207,133,250,173,248   <207>
2950 DATA207,133,251,173,247,207,141,252
,207,160,0,162,0,234,152,24,105,8,168    <105>
2960 DATA177,250,72,152,56,233,8,168,104
,145,250,200,232,224,8,208,233,206,252   <107>
2970 DATA207,240,16,230,251,165,250,24,1
05,64,133,250,144,2,230,251,76,182,23    <33>
2980 DATA173,249,207,24,105,8,141,249,20
7,144,3,238,248,207,234,234,234,234      <150>
2990 DATA234,234,234,234,234,234,234,234
,76,25,28,234,238,246,207,173,246,207    <24>
3000 DATA201,39,144,152,173,249,207,133,
250,173,248,207,133,251,173,247,207      <75>
3010 DATA141,252,207,162,0,160,0,234,189
,60,3,145,250,200,232,192,8,208,244      <214>
3020 DATA206,252,207,240,16,230,251,165,
250,24,105,64,133,250,144,2,230,251      <44>
3030 DATA76,32,24,173,219,207,141,249,20
7,133,250,173,218,207,141,248,207,133    <111>
3040 DATA251,169,0,141,246,207,169,7,141
,252,207,141,247,207,76,130,23,162,5     <241>
3050 DATA142,217,207,32,96,23,169,0,141,
23,208,141,29,208,169,7,141,21,208,169   <243>
3060 DATA1,141,28,208,169,34,141,248,7,1
62,36,142,249,7,162,60,142,250,7,162     <8>
3070 DATA42,142,251,7,162,46,142,252,7,1
62,44,142,253,7,162,43,142,254,7,162     <102>
3080 DATA47,142,255,7,169,9,141,37,208,1
69,10,141,38,208,169,8,141,39,208,169    <43>
3090 DATA9,141,40,208,169,7,141,41,208,1
69,5,141,42,208,141,46,208,169,11,141    <137>
3100 DATA43,208,169,2,141,44,208,169,9,1
41,45,208,169,0,141,0,208,141,2,208      <89>
3110 DATA169,175,141,1,208,169,196,141,3
,208,141,16,208,169,55,141,4,208,169     <136>
3120 DATA58,141,5,208,169,255,141,249,3,
32,51,20,169,3,141,249,3,169,197,141     <38>
3130 DATA9,208,141,11,208,169,191,141,7,
208,169,202,141,15,208,141,13,208,32     <170>
3140 DATA188,25,238,0,208,238,2,208,32,5
1,20,206,217,207,208,28,169,4,141,217    <14>
3150 DATA207,32,160,25,173,249,7,201,36,
240,8,169,36,141,249,7,76,67,25,169      <167>
3160 DATA35,141,249,7,173,0,208,201,130,
240,3,76,25,25,169,36,141,249,7,32,165   <154>
3170 DATA25,169,60,141,249,3,32,51,20,32
,131,25,169,37,141,248,7,32,51,20,32     <203>
3180 DATA177,25,32,51,20,169,34,141,248,
7,32,131,25,32,51,20,32,177,25,32,188    <124>
3190 DATA25,32,51,20,96,169,0,141,4,212,
141,6,212,169,8,141,5,212,169,100,141    <170>
3200 DATA0,212,169,30,141,1,212,169,33,1
41,4,212,96,173,4,212,240,6,169,0,141    <35>
3210 DATA4,212,96,169,129,141,4,212,96,1
62,6,169,0,157,0,212,202,208,250,96      <198>
3220 DATA169,17,141,5,212,169,17,141,6,2
12,169,10,141,1,212,169,69,141,0,212     <211>
3230 DATA169,0,141,4,212,96,169,0,141,33
,208,141,32,208,169,147,32,210,255,169   <104>
3240 DATA1,141,207,207,32,19,22,32,103,2
0,32,148,20,169,0,141,21,208,168,153     <149>
3250 DATA0,208,200,192,17,208,248,32,35,
22,169,70,141,250,3,169,11,141,210,207   <240>
3260 DATA169,25,141,209,207,32,44,201,23
4,234,234,234,234,169,3,141,195,207      <224>
3270 DATA169,41,133,252,169,26,133,253,7
6,229,26,150,29,29,29,17,17,17,29,29     <68>
3280 DATA29,29,29,42,32,82,32,79,32,65,3
2,68,32,82,32,85,32,78,32,78,32,69,32    <188>
3290 DATA82,32,42,95,151,17,17,29,29,29,
29,29,29,29,29,29,29,29,29,29,40,67       <230>
3300 DATA41,32,82,46,32,68,73,83,83,69,8
2,95,17,17,30,29,29,29,29,29,29,29,29     <248>
3310 DATA29,29,29,29,74,79,89,83,84,73,6
7,75,32,80,79,82,84,32,50,95,17,17,17     <132>
```

```
3320 DATA158,29,29,29,29,29,29,29,29,29,
29,29,29,29,83,67,79,82,69,32,58,95          <51>
3330 DATA17,29,29,29,29,32,29,29,29,72,7
3,71,72,45,83,67,79,82,69,32,58,95,17        <101>
3340 DATA17,17,17,159,32,32,32,32,32,32,
32,32,32,32,32,32,32,32,32,80,82,69          <78>
3350 DATA83,83,32,70,73,82,69,153,157,15
7,157,157,157,145,145,145,145,145,145        <248>
3360 DATA145,94,160,0,177,252,201,94,240
,18,201,95,208,2,169,13,32,210,255,230       <248>
3370 DATA252,208,2,230,253,76,231,26,172
,232,3,132,99,173,233,3,133,98,162,144       <27>
3380 DATA56,32,73,188,32,221,189,32,30,1
71,173,224,207,240,5,169,48,32,210,255       <198>
3390 DATA173,233,3,205,235,3,176,14,76,6
7,27,173,232,3,205,234,3,176,5,76,67         <160>
3400 DATA27,240,243,173,232,3,141,234,3,
173,233,3,141,235,3,169,13,32,210,255        <204>
3410 DATA32,210,255,162,20,169,29,32,210
,255,202,208,248,172,234,3,132,99,173        <26>
3420 DATA235,3,133,98,162,144,56,32,73,1
88,32,221,189,32,30,171,173,224,207          <179>
3430 DATA240,13,169,48,32,210,255,169,0,
141,232,3,141,233,3,169,1,141,224,207        <219>
3440 DATA234,234,234,234,234,234,32,228,
255,201,141,240,10,173,0,220,201,111         <243>
3450 DATA208,242,76,180,27,238,207,207,3
2,51,20,32,51,20,32,51,20,173,207,207        <232>
3460 DATA201,4,144,220,169,1,141,207,207
,76,136,27,173,207,207,201,1,240,12          <44>
3470 DATA201,3,240,3,76,0,192,76,21,196,
234,234,169,1,141,207,207,32,56,22,32        <234>
3480 DATA35,22,173,210,207,141,217,207,1
69,25,141,209,207,234,32,96,23,169,10        <14>
3490 DATA141,215,207,169,0,141,214,207,1
69,1,141,6,208,169,71,141,8,208,169          <133>
3500 DATA160,141,10,208,169,16,141,12,20
8,169,100,141,14,208,169,135,141,21          <7>
3510 DATA208,173,30,208,208,251,76,130,2
3,169,0,141,211,207,76,85,29,173,217         <41>
3520 DATA207,74,240,242,206,217,207,208,
118,173,0,220,201,127,240,3,76,195,28        <45>
3530 DATA234,173,30,208,41,252,240,3,76,
157,30,173,210,207,141,217,207,169,1         <7>
3540 DATA141,211,207,76,85,29,173,214,20
7,240,3,76,12,29,206,215,207,208,43          <79>
3550 DATA169,10,141,215,207,238,232,3,20
8,3,238,233,3,173,16,208,41,4,240,13         <223>
3560 DATA206,4,208,208,18,173,16,208,41,
251,141,16,208,206,4,208,173,4,208,201       <189>
3570 DATA38,144,30,173,214,207,240,3,76,
154,28,32,160,25,173,249,7,201,36,240        <243>
3580 DATA6,238,249,7,76,154,28,206,249,7
,76,3,24,32,169,28,206,210,207,206,210       <159>
3590 DATA207,76,0,192,174,209,207,224,40
,208,1,96,173,232,3,24,105,100,141,232       <165>
3600 DATA3,144,3,238,233,3,232,76,172,28
,201,126,240,21,201,125,240,17,201,123       <66>
3610 DATA240,7,201,119,240,6,76,46,28,76
,136,30,76,120,30,174,1,208,224,175          <125>
3620 DATA240,3,76,46,28,162,140,142,212,
207,201,126,240,5,169,152,141,212,207        <254>
3630 DATA169,35,141,249,7,169,2,141,214,
207,169,1,141,213,207,32,177,25,32,89        <123>
3640 DATA30,76,46,28,234,173,213,207,201
,1,240,3,76,49,29,173,1,208,205,212          <176>
3650 DATA207,144,15,206,1,208,206,1,208,
206,3,208,206,3,208,76,79,28,238,213         <124>
3660 DATA207,173,1,208,201,175,240,15,23
8,3,208,238,3,208,238,1,208,238,1,208        <174>
3670 DATA76,79,28,169,0,141,214,207,32,1
77,25,32,188,25,76,79,28,234,206,6,208       <205>
3680 DATA208,36,173,16,208,72,41,8,240,9
,104,41,247,141,16,208,76,127,29,104         <113>
3690 DATA9,8,141,16,208,169,230,141,6,20
8,173,21,208,9,8,141,21,208,206,8,208        <168>
3700 DATA208,57,173,16,208,72,41,16,240,
9,104,41,239,141,16,208,76,189,29,104        <191>
3710 DATA9,16,141,16,208,169,230,141,8,2
08,173,21,208,9,16,141,21,208,173,9          <207>
3720 DATA208,201,179,208,8,169,185,141,9
,208,76,189,29,206,9,208,238,7,208,206       <239>
3730 DATA10,208,208,36,173,16,208,72,41,
32,240,9,104,41,223,141,16,208,76,230        <37>
3740 DATA29,104,9,32,141,16,208,169,230,
141,10,208,173,21,208,9,32,141,21,208        <43>
3750 DATA206,12,208,208,46,173,16,208,72
,41,64,240,9,104,41,191,141,16,208,76        <204>
3760 DATA25,30,104,9,64,141,16,208,169,2
30,141,12,208,173,21,208,9,64,141,21         <9>
3770 DATA208,173,6,208,201,115,144,3,206
,13,208,206,14,208,208,48,173,16,208         <74>
3780 DATA72,41,128,240,9,104,41,127,141,
16,208,76,78,30,104,9,128,141,16,208         <90>
3790 DATA169,230,141,14,208,173,21,208,9
,128,141,21,208,173,4,208,201,230,144        <115>
3800 DATA5,169,190,141,15,208,173,211,20
7,240,3,76,71,28,76,31,28,169,0,141          <69>
3810 DATA4,212,169,96,141,5,212,169,0,14
1,6,212,169,0,141,0,212,169,30,141,1         <190>
3820 DATA212,169,129,141,4,212,96,173,0,
208,201,255,240,6,238,0,208,238,2,208        <93>
3830 DATA76,46,28,173,0,208,201,30,240,2
46,173,214,207,208,241,206,0,208,206         <137>
3840 DATA2,208,76,46,28,32,165,25,32,230
,30,174,209,207,160,0,140,242,207,140        <163>
3850 DATA240,207,160,22,169,59,32,164,20
,232,232,232,142,209,207,224,40,240          <153>
3860 DATA6,32,99,24,76,221,27,76,214,25,
169,0,141,4,212,162,255,141,5,212,142        <198>
3870 DATA6,212,142,0,212,142,1,212,169,5
0,141,208,207,169,17,141,4,212,96,32         <109>
3880 DATA200,30,169,4,141,249,3,238,39,2
08,238,37,208,32,51,20,206,1,212,206         <252>
3890 DATA40,208,206,208,207,208,236,169,
0,141,4,212,96,173,0,20,201,0,240,1          <129>
3900 DATA96,169,64,160,31,133,95,132,96,
169,224,160,46,133,90,132,91,169,160         <115>
3910 DATA160,207,133,88,132,89,32,191,16
3,169,1,141,0,20,96,77,80,32,78,69,84        <99>
3920 DATA71,85,0,113,31,18,112,84,69,169
,2,141,207,207,169,0,141,33,208,141          <108>
3930 DATA21,208,32,148,20,169,25,141,209
,207,32,165,25,162,0,189,40,192,32,210       <237>
3940 DATA255,232,224,39,208,245,76,79,19
2,147,17,17,17,17,17,17,29,29,29,29          <7>
3950 DATA29,29,29,152,80,82,69,83,83,32,
70,73,82,69,32,70,79,82,32,76,69,86          <82>
3960 DATA69,76,32,50,32,32,173,0,220,201
,111,208,249,32,56,22,169,1,141,200          <56>
3970 DATA207,169,137,141,242,207,32,44,2
2,169,7,141,240,207,162,0,160,17,169         <106>
3980 DATA48,32,128,192,162,0,160,19,169,
49,32,128,192,76,144,192,32,164,20,232       <9>
3990 DATA232,232,224,37,144,246,162,37,3
2,164,20,96,169,1,141,242,207,141,240        <255>
4000 DATA207,162,3,160,1,169,61,32,164,2
0,162,8,32,164,20,162,19,32,164,20,162       <233>
4010 DATA26,32,164,20,162,33,32,164,20,1
69,31,141,21,208,169,1,141,28,208,169        <237>
4020 DATA132,141,29,208,169,1,141,206,20
7,169,34,141,248,7,162,36,142,249,7          <200>
4030 DATA169,45,141,251,7,141,252,7,141,
253,7,141,254,7,141,255,7,141,250,7          <210>
```

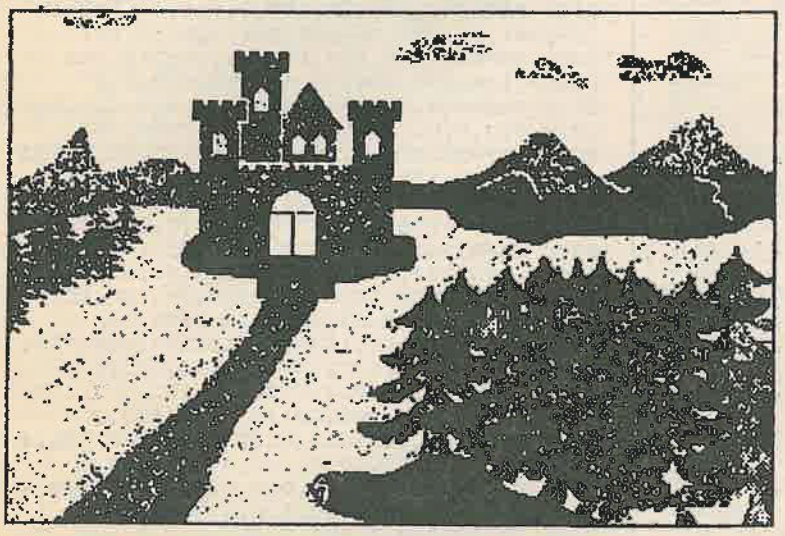
```
4040 DATA169,252,141,23,208,169,0,141,37
,208,141,16,208,169,10,141,38,208,169      <108>
4050 DATA8,141,39,208,169,0,141,40,208,1
69,14,141,42,208,141,43,208,141,44,208     <198>
4060 DATA141,45,208,141,46,208,141,41,20
8,169,130,141,4,208,169,184,141,5,208      <91>
4070 DATA141,7,208,141,9,208,141,11,208,
141,13,208,141,15,208,169,195,141,6        <93>
4080 DATA208,169,20,141,8,208,169,16,141
,16,208,169,0,141,0,208,141,2,208,141      <45>
4090 DATA204,207,169,158,141,1,208,169,1
79,141,3,208,173,30,208,208,251,32,35      <228>
4100 DATA22,32,188,25,76,131,193,32,160,
25,206,206,207,76,203,193,169,138,141      <37>
4110 DATA202,207,76,33,194,169,127,141,2
02,207,76,33,194,169,144,141,202,207       <171>
4120 DATA76,33,194,32,77,20,173,16,208,4
1,3,208,7,173,0,208,201,35,144,15,173      <201>
4130 DATA0,220,201,119,240,208,201,126,2
40,212,201,125,240,216,173,206,207,201     <97>
4140 DATA2,240,184,206,206,207,208,28,16
9,4,141,206,207,32,160,25,173,249,7        <228>
4150 DATA201,36,208,8,169,35,141,249,7,7
6,203,193,169,36,141,249,7,173,30,208      <149>
4160 DATA41,252,240,13,174,1,208,224,158
,208,6,141,201,207,76,155,194,173,0        <192>
4170 DATA208,201,80,208,10,173,16,208,41
,3,240,3,76,121,194,173,0,208,201,255      <110>
4180 DATA208,8,173,16,208,9,3,141,16,208
,238,0,208,238,2,208,173,16,208,41,3       <116>
4190 DATA208,10,173,0,208,201,33,208,3,3
2,135,194,173,204,207,208,3,76,131,193     <99>
4200 DATA76,52,194,32,89,30,169,35,141,2
49,7,169,1,141,203,207,141,204,207,76      <148>
4210 DATA203,193,32,77,20,173,203,207,20
1,1,208,23,173,1,208,205,202,207,240       <8>
4220 DATA9,206,1,208,206,3,208,76,203,19
3,238,203,207,76,203,193,173,1,208,201     <205>
4230 DATA158,240,18,201,153,144,5,169,36
,141,249,7,238,3,208,238,1,208,76,203      <221>
4240 DATA193,169,0,141,204,207,32,188,25
,76,163,193,234,169,0,141,16,208,169       <36>
4250 DATA36,141,249,7,76,85,195,234,32,1
65,25,169,36,141,249,7,32,77,25,173        <178>
4260 DATA0,220,201,111,208,249,96,173,16
,208,41,3,208,10,173,0,208,201,35,176      <129>
4270 DATA3,76,223,193,32,209,194,174,209
,207,160,0,140,242,207,140,240,207,160     <195>
4280 DATA22,169,59,32,164,20,232,232,232
,142,209,207,224,40,240,3,76,144,192       <59>
4290 DATA76,214,25,169,1,141,200,207,169
,36,141,249,7,32,165,25,173,201,207        <247>
4300 DATA41,4,208,37,173,201,207,41,8,20
8,36,173,201,207,41,16,208,35,173,201      <87>
4310 DATA207,41,32,208,34,173,201,207,41
,64,208,33,173,14,208,24,105,12,76,40      <18>
4320 DATA195,173,4,208,76,4,195,173,6,20
8,76,40,195,173,8,208,76,40,195,173        <159>
4330 DATA10,208,76,40,195,173,12,208,76,
40,195,141,0,208,141,2,208,32,200,30       <57>
4340 DATA169,4,141,249,3,238,1,208,238,3
,208,206,1,212,206,1,212,32,51,20,173      <8>
4350 DATA3,208,201,225,208,234,169,0,141
,4,212,141,21,208,96,173,232,3,24,105      <157>
4360 DATA20,141,232,3,144,3,238,233,3,23
8,200,207,173,200,207,201,2,240,55,201     <239>
4370 DATA3,240,74,201,4,240,103,201,5,24
0,66,201,6,240,39,201,7,240,120,201        <62>
4380 DATA8,240,54,201,9,240,83,201,10,24
0,46,201,11,240,104,201,12,240,71,173      <167>
4390 DATA250,3,56,233,13,141,250,3,32,16
9,28,76,21,196,169,80,141,4,208,169        <235>
4400 DATA140,141,14,208,169,135,141,21,2
08,169,0,141,16,208,76,63,193,169,70       <203>
4410 DATA141,6,208,169,130,141,8,208,169
,200,141,10,208,169,64,141,16,208,169      <195>
4420 DATA20,141,12,208,169,123,141,21,20
8,76,63,193,234,169,90,141,8,208,169       <23>
4430 DATA150,141,4,208,169,32,141,16,208
,169,40,141,10,208,169,55,141,21,208       <250>
4440 DATA76,63,193,169,100,141,4,208,169
,170,141,14,208,169,10,141,10,208,169      <179>
4450 DATA8,141,16,208,169,143,141,21,208
,76,63,193,169,3,141,207,207,32,26,22      <213>
4460 DATA169,25,141,209,207,32,148,20,32
,103,20,32,165,25,169,0,141,32,208,141     <142>
4470 DATA33,208,141,21,208,170,189,72,19
6,32,210,255,232,224,39,208,245,32,35      <117>
4480 DATA22,76,111,196,147,17,17,17,17,1
7,17,29,29,29,29,29,29,29,152,80,82        <62>
4490 DATA69,83,83,32,70,73,82,69,32,70,7
9,82,32,76,69,86,69,76,32,51,32,32,173     <69>
4500 DATA0,220,201,111,208,249,32,26,22,
32,130,20,32,56,22,32,26,22,169,0,141      <222>
4510 DATA33,208,169,136,141,242,207,169,
7,141,240,207,162,7,160,19,169,51,32       <241>
4520 DATA164,20,232,232,232,224,32,144,2
46,162,7,136,136,192,2,176,238,160,1       <248>
4530 DATA169,50,32,164,20,232,232,232,22
4,32,144,246,162,10,160,10,32,164,20       <85>
4540 DATA200,200,232,32,164,20,162,22,16
0,16,32,164,20,200,200,232,32,164,20       <135>
4550 DATA162,20,160,3,32,164,20,200,200,
232,32,164,20,169,87,141,242,207,169       <52>
4560 DATA10,141,240,207,169,69,160,19,16
2,4,32,164,20,136,136,192,2,176,247        <9>
4570 DATA160,1,32,164,20,160,19,162,32,3
2,164,20,136,136,192,2,176,247,160,1       <195>
4580 DATA32,164,20,169,0,170,157,88,33,2
32,208,250,169,67,162,4,160,19,32,164      <157>
4590 DATA20,162,32,32,164,20,162,4,160,1
2,32,164,20,162,32,32,164,20,162,4,160     <63>
4600 DATA5,32,164,20,162,32,32,164,20,32
,35,22,32,44,22,169,0,141,28,208,141       <198>
4610 DATA29,208,141,23,208,141,16,208,16
9,5,141,24,212,169,6,141,197,207,169       <215>
4620 DATA38,141,248,7,169,63,141,249,7,1
69,11,141,39,208,141,40,208,169,159        <90>
4630 DATA141,0,208,141,2,208,169,161,141
,1,208,169,140,141,3,208,169,3,141,21      <34>
4640 DATA208,32,188,25,169,0,141,41,208,
234,234,234,234,234,173,195,207,141        <102>
4650 DATA196,207,169,41,141,250,7,141,25
1,7,141,252,7,141,253,7,169,90,141,4       <8>
4660 DATA208,141,5,208,169,146,141,6,208
,141,7,208,169,200,141,8,208,141,9,208     <197>
4670 DATA169,250,141,10,208,141,11,208,1
69,0,141,41,208,141,42,208,141,43,208      <225>
4680 DATA141,44,208,141,194,207,141,193,
207,169,44,141,254,7,169,42,141,255        <86>
4690 DATA7,169,2,141,45,208,169,5,141,46
,208,169,120,141,12,208,169,225,141        <6>
4700 DATA14,208,169,100,141,15,208,173,3
0,208,208,251,160,0,140,199,207,185        <66>
4710 DATA88,58,153,184,136,200,208,247,1
69,88,133,250,169,58,133,251,169,24        <197>
4720 DATA133,252,169,57,133,253,160,0,17
7,252,145,250,200,208,249,238,199,207      <22>
4730 DATA173,199,207,201,19,208,3,76,85,
198,76,99,198,198,251,165,250,56,233       <123>
4740 DATA64,133,250,176,2,198,251,198,25
3,165,252,56,233,64,133,252,176,2,198      <62>
4750 DATA253,76,31,198,160,0,185,184,136
,153,152,34,200,208,247,76,1,198,206       <80>
```

```
4760 DATA197,207,208,46,169,6,141,197,20
7,173,248,7,201,38,240,15,201,39,240      <19>
4770 DATA22,169,38,141,248,7,32,160,25,7
6,150,198,169,39,141,248,7,32,160,25      <11>
4780 DATA76,150,198,169,40,141,248,7,32,
160,25,206,196,207,208,157,173,195,207    <78>
4790 DATA141,196,207,238,193,207,208,27,
238,194,207,173,194,207,24,105,100,141    <191>
4800 DATA232,3,144,3,238,233,3,173,194,2
07,201,19,208,3,76,188,200,173,0,220      <88>
4810 DATA201,123,240,37,201,119,240,11,2
01,126,240,51,201,125,240,63,76,35,199    <118>
4820 DATA173,0,208,201,255,240,246,238,0
,208,238,0,208,238,2,208,238,2,208,76     <175>
4830 DATA212,198,173,0,208,201,83,240,22
4,206,0,208,206,0,208,206,2,208,206       <75>
4840 DATA2,208,76,212,198,173,1,208,201,
100,240,202,206,1,208,206,3,208,76,212    <247>
4850 DATA198,173,1,208,201,170,240,186,2
38,1,208,238,3,208,76,212,198,234,173     <189>
4860 DATA30,208,240,11,41,3,240,7,201,3,
240,3,76,119,200,173,5,208,208,8,173      <184>
4870 DATA21,208,9,4,141,21,208,173,7,208
,208,8,173,21,208,9,8,141,21,208,173      <103>
4880 DATA9,208,208,8,173,21,208,9,16,141
,21,208,173,11,208,208,8,173,21,208       <130>
4890 DATA9,32,141,21,208,206,5,208,206,5
,208,208,32,173,195,207,201,2,240,9       <193>
4900 DATA173,0,208,141,4,208,76,144,199,
173,4,208,24,105,35,141,4,208,201,70      <167>
4910 DATA176,3,76,128,199,206,7,208,206,
7,208,208,16,173,6,208,56,233,85,141      <95>
4920 DATA6,208,201,70,176,3,76,152,199,2
06,9,208,206,9,208,208,16,173,8,208       <13>
4930 DATA56,233,33,141,8,208,201,70,176,
3,76,176,199,206,11,208,206,11,208,208    <25>
4940 DATA16,173,10,208,24,105,90,141,10,
208,201,70,176,3,76,200,199,173,194       <46>
4950 DATA207,201,2,176,3,76,116,200,238,
13,208,238,15,208,173,13,208,201,60       <172>
4960 DATA208,18,173,12,208,201,72,144,66
,173,21,208,9,64,141,21,208,76,56,200     <183>
4970 DATA201,210,208,51,173,21,208,41,19
1,141,21,208,173,194,207,201,12,208       <117>
4980 DATA10,169,10,141,45,208,169,46,141
,254,7,173,194,207,201,6,208,10,169       <245>
4990 DATA9,141,45,208,169,43,141,254,7,1
73,12,208,24,105,90,141,12,208,234,173    <122>
5000 DATA15,208,201,60,208,18,173,14,208
,201,72,144,44,173,21,208,9,128,141       <54>
5010 DATA21,208,76,115,200,201,210,208,2
9,173,21,208,41,127,141,21,208,173,14     <53>
5020 DATA208,56,233,90,141,14,208,173,19
4,207,201,8,208,5,169,47,141,255,7,234    <30>
5030 DATA76,56,198,174,209,207,160,0,140
,242,207,140,240,207,160,22,169,59,32     <194>
5040 DATA164,20,232,232,232,142,209,207,
224,40,240,6,32,159,200,76,127,196,32     <131>
5050 DATA159,200,76,214,25,32,200,30,162
,130,238,39,208,238,40,208,169,1,141      <103>
5060 DATA249,3,32,51,20,206,1,212,232,20
8,236,32,165,25,96,169,2,141,195,207      <188>
5070 DATA32,169,28,32,165,25,32,148,20,1
62,0,142,21,208,189,221,200,32,210,255    <252>
5080 DATA232,224,39,208,245,76,4,201,147
,17,17,17,17,17,17,29,29,29,29,29,29      <149>
5090 DATA29,152,80,82,69,83,83,32,70,73,
82,69,32,70,79,82,32,76,69,86,69,76       <169>
5100 DATA32,49,32,32,173,0,220,201,111,2
08,249,169,14,141,33,208,76,199,27,0      <134>
5110 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,120,169,0,141,11,212     <23>
5120 DATA141,18,212,169,5,141,24,212,169
,96,141,12,212,169,96,141,19,212,169      <89>
5130 DATA48,141,13,212,169,240,141,20,21
2,169,33,141,11,212,169,17,141,18,212     <194>
5140 DATA169,0,141,188,207,141,187,207,1
69,117,141,20,3,169,201,141,21,3,160      <12>
5150 DATA0,140,190,207,200,140,189,207,8
8,96,206,189,207,208,91,72,138,72,174     <45>
5160 DATA190,207,189,244,201,141,7,212,2
4,105,8,141,14,212,189,148,202,141,8      <29>
5170 DATA212,141,15,212,189,52,203,141,1
89,207,232,224,130,144,47,162,0,238       <246>
5180 DATA188,207,173,188,207,240,27,201,
1,240,10,169,255,141,188,207,169,96       <225>
5190 DATA141,19,212,169,0,141,11,212,169
,17,141,18,212,76,207,201,169,33,141      <138>
5200 DATA11,212,169,17,141,18,212,142,19
0,207,104,170,104,76,49,234,0,0,0,0       <121>
5210 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,226,0,226,0,226,133      <222>
5220 DATA55,226,0,226,0,226,133,55,0,133
,39,133,39,133,232,81,0,226,0,226,0       <213>
5230 DATA226,39,81,226,0,226,0,226,39,81
,0,196,247,196,180,55,81,133,0,226,0      <75>
5240 DATA226,133,0,133,55,0,55,196,55,0,
196,0,196,247,196,180,55,180,81,0,226     <106>
5250 DATA0,226,39,0,39,81,0,81,180,81,0,
196,247,196,180,55,81,133,0,133,207       <252>
5260 DATA133,205,133,81,133,207,226,133,
81,133,207,226,123,133,81,180,123,133     <19>
5270 DATA81,180,207,123,226,133,207,133,
205,133,81,133,207,226,133,81,133,207     <206>
5280 DATA226,123,133,81,180,123,133,81,1
80,207,123,226,0,0,0,0,0,0,0,0,0,0,0      <56>
5290 DATA0,0,0,0,0,0,0,0,0,0,0,4,0,4,0,4
,6,8,4,0,4,0,4,6,8,0,6,6,6,6,6,6,7,0      <227>
5300 DATA4,0,4,0,4,6,7,4,0,4,0,4,6,7,0,9
,10,9,8,8,7,6,0,4,0,4,6,0,6,8,0,8,9       <22>
5310 DATA8,0,9,0,9,10,9,8,8,8,7,0,4,0,4,
6,0,6,7,0,7,8,7,0,9,10,9,8,8,7,6,0,6      <6>
5320 DATA5,6,5,6,7,6,5,4,6,7,6,5,4,5,6,7
,8,5,6,7,8,5,5,4,6,5,6,5,6,7,6,5,4,6      <117>
5330 DATA7,6,5,4,5,6,7,8,5,6,7,8,5,5,4,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0      <127>
5340 DATA0,0,0,6,1,6,1,6,18,12,6,1,6,1,6
,18,30,6,12,6,6,6,6,6,24,6,6,1,6,1,6      <71>
5350 DATA18,12,6,1,6,1,6,18,30,6,12,6,6,
6,6,6,48,12,6,1,6,6,1,6,6,1,6,18,30       <91>
5360 DATA12,6,1,6,6,6,6,6,18,30,12,6,1,6
,6,1,6,6,1,6,18,30,12,6,6,6,6,6,6,24      <181>
5370 DATA6,24,24,24,24,12,12,12,12,24,12
,12,12,12,48,24,24,12,12,24,24,12,12      <155>
5380 DATA24,24,48,24,24,24,24,12,12,12,1
2,24,12,12,12,12,48,24,24,12,12,24,24     <223>
5390 DATA12,12,24,24,48,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0      <169>
5400 DATA0,0,0,0,0,0,0,0                  <4>
5410 :                                    <113>
5420 :                                    <123>
5430 :                                    <133>
5440 PRINT"{DOWN SPACE}PASS{SPACE}1"      <88>
5450 FORI=2048TO11039:READA:B=B+A:NEXT    <11>
5460 IFB<>1008209THENPRINT"{DOWN SPACE}F
EHLER{SPACE}IN{SPACE}DATAS":END           <71>
5470 :                                    <173>
5480 RESTORE                              <9>
5490 PRINT"{DOWN SPACE}PASS{SPACE}2"      <148>
5500 FORI=2048TO11039:READA:POKEI,A:NEXT  <197>
5510 PRINT"{DOWN SPACE}PROGRAMM{SPACE}JE
TZT{SPACE}ABSPEICHERN"                    <127>
5520 POKE46,44                            <50>
  ENDE DES LISTINGS
```

Was gibt's Neues auf dem Software-Markt

Hi-EDDI plus

Fantastisches Zeichenprogramm für den C-64



Mit Hi-EDDI plus stellt Markt & Technik ein Programm vor, das nicht wie üblich als Fertigprodukt aus einem großen Softwarehaus stammt, sondern von einem Anwender, Conputerfan und Student der Elektrotechnik zunächst nur für eigene Zwecke entwickelt wurde.

Der erste Erfolg stellte sich ein, als das Programm in einer Computerzeitung veröffentlich wurde. Seitdem meldeten sich bei dem Autor eine Vielzahl von Anwendern mit neuen Tips und auch Wünschen, die Hans Haberl für eine Vielzahl von Verbesserungen und Erweiterungen verwertete. Das Ergebnis ist HI-EDDI plus, ein Malprogramm mit vielen ganz besonderen Eigenschaften. HI-EDDI + nutzt zunächst einmal konsequent den High-Resolutionmodus, um die größtmöglichste Bildauflösung auf dem C-64 zu erreichen. Damit sind mit HI-EDDI technische Zeichnungen kein Problem mehr.

Technisches Zeichnen wird zudem noch unterstützt durch Construktion-Sets, die zum Beispiel Schaltbildsymbole enthalten, und der Möglichkeit, bis zu 6 Bildschirme zu einem Großbildschirm zusammenzufassen und nahtlos ausdrucken zu lassen. Wollten wir alle Vorzüge dieses Zeichenprogramms aufführen, wäre der Rahmen dieser Besprechung sicher gesprengt.

Auf zwei besondere Dinge sei jedoch hingewiesen. HI-EDDI + besticht vor allem durch seine exzellente Dokumentation in Buchform, die unserer Meinung nach der beste Schutz gegen unerlaubtes Kopieren ist. Die 48,-- DM lohnen sich schon allein für die fantastische und sehr verständlich geschriebene Anleitung, die auch Wortindex, Befehlsübersicht und ausführliche Assemblerdokumentation beinhaltet, was man bei vielen Programmen vergeblich sucht. Besonders gefallen haben uns auch die Runtime-Module, mit denen man die fertigen Bilder auch in eigene Programme einbinden kann, auch in eigene Grafikadventure! HI-EDDI + besitzt Druckertreiber für die meisten gängigen Drucker und arbeitet mit Joystick und Koala-Pad als Eingabegeräte zusammen. (uk)

HI-EDDI plus, Markt & Technik, 8013 Haar, Preis: 48,- DM

NEU für den Schneider CPC

BASIC COMPILER V2.0

Das Softwareteam T. Brandt und F. Thilen hat eine verbesserte Version ihres Basic-Compilers auf den Markt gebracht. Die Version 2.0 vereinigt alle Features seines Vorgängers in sich, wurde aber in einer entscheidenden Sache verbessert: Speicherplatz ist nun auch für längere Programme vorhanden. Mit der neuen Version können nun auch eigenständige Programme entwickelt werden, wobei allerdings der Nachteil der vom Locomotive Basic abweichenden Syntax nach wie vor vorhanden ist. Ebenfalls besteht die Möglichkeit, fast jedes beliebige Textprogramm zur Programmerstellung zu verwenden, allerdings fällt dann die Möglichkeit weg, die Syntax eines Programmes vor dessen Compilierung testen zu lassen. Alles in allem ein Programm, das hauptsächlich zu Optimierung und Geschwindigkeitssteigerung bereits bestehender Programme eingesetzt werden kann. Für den Software-Entwickler sicher nicht uninteressant.

Vertrieb: Software Team
Joachim Guenster
Mühlenstr. 12
5431 Boden

Das Soft- und Hardware-Angebot für C-16/116 wurde erweitert

In den letzten Wochen wurde das Soft-und Hardware Angebot der Fa. Kingsoft erheblich erweitert. So wurden die Spielprogramme

Bongo Construction Set	**Preis 25,-- DM**
Galaxy	**Preis 19,-- DM**
Ghost Town	**Preis 19,-- DM**
Space Pilot	**Preis 25,-- DM**
Tom	**Preis 25,-- DM**
Winter Olympiade	**Preis 29,-- DM**

neu mit in das Liefersortiment aufgenommen. Die Programme Galaxy, Ghost Town, Grandmaster und Tom sind auch als Sammelkassette erhältlich und kosten im PLUS-Paket zusammen 39,-- DM.
Alle Programme sind mit eingebautem Turbotape auf Kassette abgespeichert, so daß die Ladezeit der einzelnen Programme deutlich verkürzt wird. Sie beträgt teilweise nur 1 Minute oder liegt noch darunter.

Das Programm Winter Olympiade lag der Redaktion leider nur als Demo-Version vor. Aber der kurze Einblick in dieses Programm, läßt auf einen absoluten Knüller hoffen. So wird das Spiel insgesamt 6 Disziplinen beinhalten (Abfahrt, Biathlon, Bobfahren, Eissschnellauf, Skispringen, Slalom) und für 4 Mitspieler ausgelegt werden. Die Eröffnungszeremonie, die Namenseingabe, die Landesflaggen und die Hymnen werden selbstverständlich nicht fehlen. Der Aufbau ist also ähnlich wie bei den Winter Games angelegt.

Als weiteres sind auch einige Utilities in dem Programmangebot enthalten. So wird u. a. ein

Grafik-Designer **Preis 19,-- DM**
(Zeichensatzeditor mit Animation)

Micro-Datei **Preis 29,-- DM**
Dateiverwaltung in freiem Maskenaufbau

Micro Kalk **Preis 29,-- DM**
(Tabellenkalkulation, einfach zu bedienen)

Micro Text **Preis 29,-- DM**
Textverarbeitung u.a. mit Blocksatz)

Music Master **Preis 29,-- DM**
(Musik-Pgm. mit Synthes. & Sequenzer)

Paintbox **Preis 25,-- DM**
(Mal- u. Zeichenpgm. ähnlich Koala Paint)

Turbo-Tape **Preis 19,-- DM**
(8mal schneller laden und speichern)

geliefert. Wer also einige nützliche Anwenderroutinen benötigt, sollte auf diese Angebote zurückgreifen.

Im Hardware-berich werden Joysticks, Joystick-Adapter, Speichererweiterungen u.v.m. angeboten, die zu einem sinnvollen Ausbau Ihres Systems beitragen.

Bezugsadresse: Kingsoft, Schnackebusch 4, 5106 Roetgen

Betrifft: Autorenrechte

Aus gegebenem Anlaß müssen wir auf eine besondere Form des Raubkopierens hinweisen. Im Verlag mußten wir wiederholt feststellen, daß unter den eingesandten Programmen unserer Leser Software war, deren Urheberrechte nicht oder nicht mehr bei den Einsendern lagen. In einigen Fällen hatten die Autoren ihr Programm schon in anderen Computerzeitungen veröffentlicht und Honorar kassiert. In einem anderen Fall hat ein Leser das kopierte Programm eines Bekannten eingeschickt und Autorenrechte geltend gemacht. Diese Form des Raubkopierens kann böse Folgen für die Einsender, bei Minderjährigen auch für die Eltern, haben. Nicht immer sind die eigentlichen Autoren bereit, einem außergerichtlichen Vergleich zuzustimmen. Sie werden sich jedoch in erster Linie an den Verlag wenden, der wiederum sich an dem Übeltäter schadlos halten muß. Anders als beim unerlaubten Kopieren von geschützter Software ist der Kopierer von Programmeinsendungen nicht durch Anonymität geschützt und muß in den allermeisten Fällen mit Strafverfolgung rechnen. Im Interesse unserer Leser und um unnötigen Ärger vor allem für Jugendliche zu vermeiden, bitten wir Programmeinsender, nur Programme zu schicken, die in allen wesentlichen Teilen aus eigener Feder stammen und uns dies in dem Coupon für Programmautoren mit einer separaten Unterschrift zu bestätigen.

Merke!
Keine Software-Unterlagen einsenden, wenn man nicht in besitz der Urheberrechte ist, um unnötigen Ärger zu vermeiden.

Achtung . . .

Die nächste C-16/116-Sonderausgabe ist ab

16. Juni 1986

überall im Zeitschriftenhandel erhältlich.

Versäumen Sie nicht diese Ausgabe!

Es gibt eine Riesen-Gewinn-Chance und wie immer jede Menge „Futter" für Ihr System.

Schickt uns Eure Assemblerroutinen!

Basic-Extension für Commodore 64

Tolles Programm, um Befehlserweiterungen selbst zu erstellen

Thomas Lais hat uns ein sehr gutes Programm geschickt, um kurze Assemblerroutinen mit Basicbefehlswörtern aufzurufen. Unser Appell an unsere Leser: Schickt uns Eure kurzen Assemblerprogramme, damit sie von den anderen Lesern mit Basicextension in eine Befehlserweiterung eingebunden werden können. Im folgenden erklärt Thomas Lais selbst den Aufbau und die Bedienung seines Programms.

Ein großer Minuspunkt des C-64 ist sein recht mageres Basic.

Dem kann man abhelfen, indem man Basicerweiterungen verwendet – seien es käufliche oder in Zeitschriften veröffentlichte. Aber auch das ist nicht immer die beste Lösung, denn wahrscheinlich fehlt gerade der Befehl, den man für ein bestimmtes Programm braucht. Also bleibt nur eine Basicerweiterung selbst zu schreiben – und steht dann vor dem Problem, diese einzelnen Befehle aufzurufen. **SYS** Aufrufe zu verwenden, führt mit Sicherheit zu Fehlern, wenn 2 Adressen verwechselt werden. Am besten ist es doch, den Aufruf mittels eines Befehlswortes zu vollziehen. Und dieses Programm hilft dabei! Sie benötigen nur das oder die Assemblerprogramme, die dann mittels Befehlsworte aufgerufen werden sollen, die auf Diskette oder Kassette vorliegen müssen. Das Programm muß dabei bereits an der Stelle liegen, an der es auch in der Befehlserweiterung liegen soll.

Nach Starten des Programms können Sie wählen zwischen der Neuerstellung einer Basicerweiterung und der Bearbeitung einer bereits vorhandenen Erweiterung. Zunächst zum Punkt Neuerstellung. Das Programm fragt die Startadresse und die Endadresse der Erweiterung ab und überprüft die Eingaben. Erlaubt sind nur die Bereiche über **12800** (darunter liegt das Programm) bis **40959** (ab da beginnt der Basicinterpreter) oder von **49152** bis **53247**. Danach wird das Assemblerprogramm, das die Dekodierung der Befehlsworte und den Aufruf der einzelnen Routinen vornimmt, in den Bereich ab der Startadresse gepoked. Ebenso die Datas für das fest installierte Befehlswort **OUT** und die Datas für diesen Befehl, der ein Ausschalten der Erweiterung bewirkt. Die Datas sind für eine Startadresse ab 49152 ausgelegt. Wurde eine andere Startadresse gewählt, wird das Assemblerprogramm noch entsprechend verändert. Nun ist die Befehlserweiterung initialisiert und Menü 2 wird angesprungen.

Im Menü 2 stehen 5 Möglichkeiten zur Wahl: Anzeige bisheriger Befehle, Befehl löschen, Basicerweiterung abspeichern, Laden eines neuen Befehls und Ende.

Anzeige bisheriger Befehle zeigt die bisher vorhandenen Befehle samt der Startadressen an. Befehl löschen streicht einen Befehl aus der Befehlsliste. Der Befehl **OUT** ist fest eingebaut und kann nicht gelöscht werden. **Saven** der Befehlserweiterung sichert das gesamte Assemblerprogramm auf Diskette oder Kassette.

Beim Laden eines neuen Befehls wird zunächst überprüft, ob noch Platz ist in der Erweiterung. Es können maximal 50 Befehle pro Befehlserweiterung verwendet werden, wobei die Gesamtzahl der Befehlsbuchstaben kleiner als 270 sein muß. Nach Eingabe des Namens und des Gerätes wird das Assemblerprogramm geladen. Die Startadresse und die Endadresse werden automatisch bestimmt. Falls diese beiden Adressen in einem falschen Bereich liegen, meldet das Basicprogramm das. War aber alles in Ordnung, wird nun das neue Befehlswort eingegeben und eingebunden. ACHTUNG: Das Programm erkennt nicht, ob der neu geladene Befehl einen bereits vorhandenen ganz oder teilweise überschreibt. Das aber führt dazu, daß es bei einem Aufruf des überschriebenen Befehls zu einem Programmabsturz kommt. Vermeiden Sie dies, indem Sie Aufzeichnungen über die Länge und die Lage der einzelnen Assemblerprogramme führen.

„Ende" beendet das Programm. Wurde eine Veränderung am Assemblerprogramm vorgenommen, also ein Befehl gelöscht oder geladen, ohne das die geänderte Version gespeichert wurde, kommt es noch zu einer Sicherheitsabfrage ob das Programm auch wirklich beendet werden soll. Bei Eingabe von **N** wird zum Menü 2 gesprungen.

Wird einer der 3 Punkte **„Befehl löschen"**, **„Befehl laden"** oder **„Saven"** versehentlich angewählt, kann durch die Eingabe von „←" bei der ersten vom Programm erwarteten Eingabe zum Menü 2 zurückgesprungen werden.

Wurde bei Menü 1 der Punkt **„Bearbeiten einer Basicerweiterung"** gewählt, werden der Name der Erweiterung und das Gerät abgefragt, dann die Erweiterung geladen. Die Startadresse und die bisherige Endadresse werden automatisch bestimmt und ausgegeben; dann wird die Eingabe einer neuen Endadresse erwartet. Diese muß größer oder gleich der bisherigen Endadresse sein.

Wird das Programm aus irgendwelchen Gründen unterbrochen, kann es mit **GOTO 3050** ohne Datenverlust fortgesetzt werden.

Die abgespeicherte Befehlserweiterung wird mit **LOAD "NAME",8,1** oder **,1,1** geladen

und initialisiert mittels **SYS SA**, wobei **SA** die Startadresse ist, ab der das Programm liegt.
Liegt die Erweiterung im Basicspeicher, also unterhalb **40959**, so muß vor dem Laden der Basicspeicher entsprechend beschränkt werden.

```
10 REM ********************************    <15>
20 REM ***                          ***    <17>
30 REM ***     BASIC-EXTENSION      ***    <11>
40 REM *** ======================== ***    <39>
50 REM ***                          ***    <47>
60 REM ***        (C) 1986 BY       ***    <200>
70 REM ***         ARTEXSOFT        ***    <157>
80 REM ***                          ***    <77>
90 REM ********************************    <95>
100 IF A=0 THEN POKE 52,50:POKE56,50:CLR    <91>
110 IF A=1 GOTO 7270                        <18>
120 IF A=2 GOTO 2160                        <243>
130 DEF FNL(X)=X-(INT(X/256)*256)           <104>
140 DEF FNH(X)=INT(X/256)                   <195>
200 LL$="{SPACE39}"                         <100>
300 FOR I=710 TO 735                        <88>
310 :  READ P:POKE I,P                      <180>
320 NEXT                                    <195>
350 DATA162,8,160,1,32,186,255,169,255,1
62,224,160,2,32,189,255,169,251,162         <54>
360 DATA0,160,0,32,216,255,96               <20>
700 REM *******                             <11>
730 PRINTCHR$(147)                          <172>
740 PRINT"{RVSON}*****{SPACE}HAUPTMENUE{
SPACE}*****{RVSOFF}"                        <38>
750 PRINT:PRINT:PRINT"BITTE{SPACE}WAEHLE
N:"                                         <200>
760 PRINT:PRINT:PRINT"(1){SPACE}NEUERSTE
LLUNG{SPACE}BASICERWEITERUNG"               <126>
770 PRINT:PRINT"(2){SPACE}BEARBEITUNG{SP
ACE}BASICERWEITERUNG"                       <71>
780 GET A$:IF A$="2" GOTO 2000              <81>
790 IF A$<>"1" GOTO 780                     <165>
1000 REM                                    <122>
1010 REM ***** NEUERSTELLUNG *****          <223>
1020 REM                                    <142>
1050 PRINTCHR$(147)                         <238>
1060 PRINT"{SPACE RVSON}***{SPACE}NEUERS
TELLUNG{SPACE}BASICERWEITERUNG{SPACE}***
{RVSOFF}"                                   <197>
1070 PRINT:PRINT:PRINT                      <239>
1080 INPUT"STARTADRESSE";SA                 <6>
1090 PRINT:PRINT                            <30>
1100 INPUT"ENDADRESSE{SPACE2}";EA           <19>
1110 IF SA>=EA GOTO 1180                    <132>
1120 IF (SA>12800 AND SA<40960 AND EA<40
960) GOTO 1200                              <244>
1130 IF (SA>49151 AND SA<13*4096 AND EA<
13*4096) GOTO 1200                          <82>
1180 PRINT:PRINT:PRINT"FALSCHE{SPACE}EIN
GABEN.{SPACE}BITTE{SPACE}KORRIGIEREN"       <198>
1190 FOR I=1 TO 2000:NEXT:GOTO 1000         <172>
1200 :                                      <238>
1210 FOR I=SA TO SA+143:READ P:POKEI,P:N
EXT                                         <96>
1220 DATA169,11,141,8,3,169,192,141,9,3,
96,32,115,0,144,120,201,90,176,116          <47>
1230 DATA201,65,144,111,141,139,192,162,
0,142,140,192,169,192,133,252,169           <189>
1240 DATA242,133,251,160,0,238,140,192,1
77,251,48,35,209,122,208,4,200,76           <112>
1250 DATA45,192,173,140,192,205,141,192,
240,67,200,177,251,16,251,200,24,152        <189>
1260 DATA101,251,133,251,144,2,230,252,7
6,40,192,41,127,209,122,208,34,200          <125>
1270 DATA24,152,101,122,133,122,144,2,23
0,123,173,140,192,10,170,189,140,192        <116>
1280 DATA141,119,192,189,141,192,141,120
,192,32,0,194,76,234,167,173,140,192        <79>
1290 DATA205,141,192,208,194,173,139,192
,56,76,231,167,1,1,1,0,194                  <212>
1300 FOR I=SA+242 TO SA+244:READ P:POKE
I,P:NEXT                                    <9>
1310 DATA79,85,212                          <54>
1320 FOR I=SA+512 TO SA+522:READ P:POKE
I,P:NEXT                                    <6>
1330 DATA169,228,141,8,3,169,167,141,9,3
,96                                         <154>
1400 IF SA=49152 GOTO 1800                  <121>
1500 POKE SA+1,FNL(SA+11):POKE SA+6,FNH(
SA+11)                                      <208>
1510 POKE SA+25,FNL(SA+139):POKE SA+26,F
NH(SA+139)                                  <91>
1520 POKE SA+30,FNL(SA+140):POKE SA+31,F
NH(SA+140)                                  <89>
1530 POKE SA+33,FNL(SA+242):POKE SA+37,F
NH(SA+242)                                  <151>
1540 POKE SA+43,FNL(SA+140):POKE SA+44,F
NH(SA+140)                                  <228>
1550 POKE SA+55,FNL(SA+45):POKE SA+56,FN
H(SA+45)                                    <195>
1560 POKE SA+58,FNL(SA+140):POKE SA+59,F
NH(SA+140)                                  <170>
1570 POKE SA+61,FNL(SA+141):POKE SA+62,F
NH(SA+141)                                  <48>
1580 POKE SA+82,FNL(SA+40):POKE SA+83,FN
H(SA+40)                                    <240>
1590 POKE SA+102,FNL(SA+140):POKE SA+103
,FNH(SA+140)                                <7>
1600 POKE SA+107,FNL(SA+140):POKE SA+108
,FNH(SA+140)                                <182>
1610 POKE SA+110,FNL(SA+119):POKE SA+111
,FNH(SA+119)                                <60>
1620 POKE SA+113,FNL(SA+141):POKE SA+114
,FNH(SA+141)                                <154>
1630 POKE SA+116,FNL(SA+120):POKE SA+117
,FNH(SA+120)                                <108>
```

```
1640 POKE SA+125,FNL(SA+140):POKE SA+126
,FNH(SA+140)                                <218>
1650 POKE SA+128,FNL(SA+141):POKE SA+129
,FNH(SA+141)                                <124>
1660 POKE SA+133,FNL(SA+139):POKE SA+134
,FNH(SA+139)                                <117>
1670 POKE SA+142,FNL(SA+512):POKE SA+143
,FNH(SA+512)                                <104>
1800 PRINT:PRINT:PRINT                      <204>
1810 PRINT"{SPACE}DIE{SPACE}BASICERWEITE
RUNG{SPACE}IST{SPACE}INITIALISIERT"         <121>
1900 PRINT:PRINTTAB(10)"ZUM{SPACE}MENUE{
SPACE}MIT{SPACE RVSON}SPACE{RVSOFF}"        <206>
1910 GET A$:IF A$<>"{SPACE}" GOTO 1910      <132>
1920 GOTO 3050                              <202>
2000 REM                                    <102>
2010 REM ***** LADEN BASICERW *****         <132>
2020 REM                                    <122>
2050 PRINTCHR$(147)                         <218>
2060 PRINT"{SPACE3 RVSON}*****{SPACE}LAD
EN{SPACE}BASICERWEITERUNG{SPACE}*****{RV
SOFF}"                                      <176>
2080 PRINT:PRINT:INPUT"NAME{SPACE}DER{SP
ACE}ERWEITERUNG";NA$                        <212>
2090 IF NA$="" GOTO 2050                    <81>
2100 IF NA$="_" GOTO 730                    <13>
2110 PRINT:INPUT"{RVSON}K{RVSOFF}ASSETTE
/{RVSON}D{RVSOFF}ISKETTE";GE$               <238>
2120 IF GE$<>"D" AND GE$<>"K" GOTO 2050     <41>
2130 IF GE$="D" THEN GE=8                   <208>
2140 IF GE$="K" THEN GE=1                   <176>
2150 A=2:LOAD NA$,GE,1                      <1>
2160 IFGE=1THENSA=PEEK(829)+PEEK(830)*25
6:EA=PEEK(831)+PEEK(832)*256-1:GOTO2210     <154>
2170 EA=PEEK(174)+PEEK(175)*256-1           <63>
2180 OPEN 2,8,2,NA$+",P,R"                  <200>
2190 GET#2,A$:GET#2,B$:CLOSE 2              <41>
2200 SA=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*
256                                         <165>
2210 PRINT:PRINT:PRINT"BASICERWEITERUNG{
SPACE RVSON}"NA$"{RVSOFF SPACE}GELADEN"     <148>
2220 PRINT:PRINT"STARTADRESSE:"SA           <247>
2230 PRINT:PRINT"BISHERIGE{SPACE}ENDADRE
SSE:"EA                                     <170>
2240 PRINT:INPUT"NEUE{SPACE}ENDADRESSE";
FF                                          <79>
2250 IF FF<EA GOTO 2240                     <143>
2260 PRINT:PRINT"BASICERWEITERUNG{SPACE}
BETRIEBSBEREIT"                             <160>
2270 PRINT:PRINT:PRINTTAB(10)"ZUM{SPACE}
MENUE{SPACE}MIT{SPACE RVSON}SPACE{RVSOFF
}"                                          <168>
2280 GET A$:IF A$="{SPACE}" GOTO 3050       <214>
2290 GOTO 2280                              <77>
3000 REM                                    <82>
3010 REM ***** BEARBEITEN *****             <247>
3020 REM                                    <102>
3050 PRINTCHR$(147)                         <197>
3060 PRINT"{SPACE RVSON}****{SPACE}BEARB
EITUNG{SPACE}BASICERWEITERUNG{SPACE}****
{RVSOFF}"                                   <122>
3070 PRINT:PRINT"BITTE{SPACE}WAEHLEN:"      <200>
3080 PRINT:PRINT                            <236>
3100 PRINT"(1){SPACE}ANZEIGE{SPACE}BISHE
RIGE{SPACE}BEFEHLE"                         <120>
3110 PRINT:PRINT"(2){SPACE}BEFEHL{SPACE}
LOESCHEN"                                   <221>
3120 PRINT:PRINT"(3){SPACE}BASICERWEITER
UNG{SPACE}SAVEN"                            <230>
3130 PRINT:PRINT"(4){SPACE}LADEN{SPACE}N
EUEN{SPACE}BEFEHLS"                         <215>
3140 PRINT:PRINT"(5){SPACE}ENDE"            <48>
3200 GET A$:IF A$<"1" OR A$>"5" GOTO 320
0                                           <4>
3220 ON ASC(A$)-48 GOTO 4000,5000,6000,7
000,8000                                    <2>
4000 REM                                    <62>
4010 REM ***** ANZEIGE *****                <187>
4020 REM                                    <82>
4050 PRINTCHR$(147)                         <177>
4060 PRINT"{SPACE RVSON}*****{SPACE}ANZE
IGEN{SPACE}BISHERIGE{SPACE}BEFEHLE{SPACE
}*****{RVSOFF}"                             <223>
4100 I=1:J=5:K=SA+242:L=SA+142              <217>
4110 POKE 214,J:PRINT:D$=""                 <158>
4120 D$=D$+CHR$(PEEK(K)AND127)              <106>
4130 IF (PEEK(K)AND128)=0 THEN K=K+1:GOT
O 4120                                      <107>
4140 Z=PEEK(L)+PEEK(L+1)*256:K=K+1          <10>
4150 PRINTD$TAB(25)Z                        <134>
4170 I=I+1:J=J+1:L=L+2                      <128>
4180 IF I>PEEK(SA+141) GOTO 4300            <47>
4190 IF J<20 GOTO 4110                      <33>
4200 POKE 214,22:PRINT:PRINTTAB(8)"WEITE
R{SPACE}MIT{SPACE RVSON}SPACE{RVSOFF}"      <47>
4210 GET A$:IF A$<>"{SPACE}" GOTO 4210      <74>
4220 FOR T=5 TO 22:POKE 214,T:PRINT:PRIN
TLL$:NEXT                                   <147>
4260 J=5:GOTO 4110                          <93>
4300 POKE214,22:PRINT:PRINTTAB(10)"ZUM{S
PACE}MENUE{SPACE}MIT{SPACE RVSON}SPACE{R
VSOFF}"                                     <51>
4310 GET A$:IF A$<>"{SPACE}" GOTO 4310      <189>
4320 GOTO 3000                              <31>
5000 REM                                    <42>
5010 REM ***** BEFEHL LOESCHEN *****        <67>
5020 REM                                    <62>
5050 PRINTCHR$(147)                         <157>
5060 PRINT"{SPACE5 RVSON}*****{SPACE}BEF
EHL{SPACE}LOESCHEN{SPACE}*****{RVSOFF}"     <0>
5100 PRINT:PRINT:INPUT"ZU{SPACE}LOESCHEN
DER{SPACE}BEFEHL";BE$                       <255>
5130 IF BE$="" GOTO 5050                    <76>
5140 IF BE$="_" GOTO 3050                   <80>
5170 IFBE$="OUT"THENPRINT:PRINT"GEHT{SPA
CE}NICHT!":FORT=1TO3000:NEXT:GOTO5050       <126>
5180 BN$=LEFT$(BE$,LEN(BE$)-1)+CHR$(ASC(
RIGHT$(BE$,1)) OR 128)                      <182>
5200 I=1:J=SA+242:K=1                       <131>
5210 L=J                                    <252>
5220 IF ASC(MID$(BN$,I,1))=PEEK(J) GOTO
5400                                        <120>
5230 IF PEEK(SA+141)<K GOTO 5300            <29>
5240 IF (PEEK(J)AND128)=0 THEN J=J+1:GOT
O 5240                                      <253>
5250 J=J+1:I=1:K=K+1:GOTO 5210              <31>
5300 PRINT:PRINT:PRINT"DIESER{SPACE}BEFE
HL{SPACE}IST{SPACE}HIER{SPACE}NICHT{SPAC
E}VORHANDEN!"                               <227>
5310 PRINT:PRINTTAB(10)"ZUM{SPACE}MENUE{
SPACE}MIT{SPACE RVSON}SPACE{RVSOFF}"        <45>
5320 GET A$:IF A$="{SPACE}" GOTO 3050       <194>
5330 GOTO 5320                              <42>
5400 IF (PEEK(J)AND 128)=128 GOTO 5500      <84>
5420 J=J+1:I=I+1                            <165>
5430 IF I<=LEN(BN$) GOTO 5220               <153>
5440 GOTO 5240                              <158>
5500 IF K=PEEK(SA+141) GOTO 5650            <238>
5510 I=J+1:M=K                              <109>
5520 IF (PEEK(I)AND128)=0 THEN I=I+1:GOT
O 5520                                      <6>
5530 IF PEEK(SA+141)>M THEN M=M+1:I=I+1:
GOTO 5520                                   <240>
5540 FOR T=L TO I+L-J-1                     <164>
```

```
5550 :  POKE T,PEEK(T+J+1-L)                                  <164>
5560 NEXT                                                     <79>
5570 FOR T=SA+140+2*K TO SA+139+2*PEEK(S
A+141)                                                        <48>
5580 :  POKE T,PEEK(T+2)                                      <167>
5590 NEXT                                                     <109>
5650 POKE SA+141,PEEK(SA+141)-1                               <129>
5660 ZZ=1                                                     <26>
5670 PRINT:PRINT"BEFEHL{SPACE}GELOESCHT.
":PRINT:PRINTTAB(10)"ZUM{SPACE}MENUE{SPA
CE}MIT{SPACE RVSON}SPACE{RVSOFF}"                             <83>
5680 GET A$:IF A$="{SPACE}" GOTO 3050                         <44>
5690 GOTO 5680                                                <181>
6000 REM                                                      <22>
6010 REM ***** SAVEN *****                                    <67>
6020 REM                                                      <42>
6050 PRINTCHR$(147)                                           <137>
6060 PRINT"{SPACE RVSON}*****{SPACE}SAVE
N{SPACE}BEFEHLSERWEITERUNG{SPACE}*****{R
VSOFF}"                                                       <9>
6090 PRINT:PRINT:INPUT"PROGRAMMNAME";PN$                      <63>
6100 IF PN$="_" GOTO 3050                                     <74>
6110 IF PN$="" OR LEN(PN$)>16 GOTO 6050                       <197>
6120 PRINT:PRINT:INPUT"{RVSON}K{RVSOFF}A
SSETTE/{RVSON}D{RVSOFF}ISKETTE";GE$                           <50>
6130 IF GE$<>"K" AND GE$<>"D" GOTO 6050                       <243>
6140 IF GE$="K" THEN GE=1                                     <95>
6150 IF GE$="D" THEN GE=8                                     <148>
6200 POKE 711,GE                                              <21>
6210 POKE 718,LEN(PN$)                                        <193>
6220 POKE 251,FNL(SA):POKE 252,FNH(SA)                        <116>
6230 POKE 729,FNL(EA+1):POKE 731,FNH(EA+
1)                                                            <157>
6240 FOR T=1 TO LEN(PN$):POKE 735+T,ASC(
MID$(PN$,T,1)):NEXT                                           <199>
6260 SYS 710                                                  <235>
6320 ZZ=0                                                     <172>
6500 PRINT:PRINT:PRINT"SAVEN{SPACE}BEEND
ET{SPACE}!"                                                   <171>
6510 PRINT:PRINT:PRINTTAB(8)"ZUM{SPACE}M
ENUE{SPACE}MIT{SPACE RVSON}SPACE{RVSOFF}
"                                                             <189>
6520 GET A$:IF A$<>"{SPACE}" GOTO 6520                        <178>
6530 GOTO 3050                                                <222>
7000 REM                                                      <2>
7010 REM ***** LADEN NEUER BEFEHL *****                       <158>
7020 REM                                                      <22>
7050 PRINTCHR$(147)                                           <117>
7060 PRINT"{SPACE RVSON}*****{SPACE}LADE
N{SPACE}NEUER{SPACE}BEFEHL{SPACE}*****{R
VSOFF}"                                                       <71>
7090 IF PEEK(SA+141)=50 GOTO 7160                             <122>
7100 J=SA+242                                                 <252>
7110 FOR I=1 TO PEEK(SA+141)                                  <113>
7120 IF(PEEK(J) AND 128)=0 THEN J=J+1:GO
TO7120                                                        <65>
7130 J=J+1                                                    <30>
7140 NEXT                                                     <129>
7150 IF SA+511-J>3 GOTO 7190                                  <116>
7160 PRINT:PRINT"KEIN{SPACE}PLATZ{SPACE}
MEHR!":PRINT:PRINT"ZUM{SPACE}MENUE{SPACE
}MIT{SPACE RVSON}SPACE{RVSOFF}"                               <199>
7170 GET A$:IF A$="{SPACE}" GOTO 3050                         <4>
7180 GOTO 7170                                                <126>
7190 PRINT:PRINT:INPUT"ZU{SPACE}LADENDER
{SPACE}BEFEHL";PN$                                            <53>
7200 IF PN$="_" GOTO 3050                                     <155>
7210 IF PN$="" GOTO 7050                                      <189>
7220 PRINT:INPUT"{RVSON}K{RVSOFF}ASSETTE
/{RVSON}D{RVSOFF}ISKETTE";GE$                                 <248>
7230 IF GE$<>"D" AND GE$<>"K" GOTO 7220                       <127>
7240 IF GE$="D" THEN GE=8:GOTO 7260                           <43>
7250 GE=1                                                     <25>
7260 A=1:LOAD PN$,GE,1                                        <111>
7270 IF GE=1 THENEN=PEEK(831)+PEEK(832)*
256:SS=PEEK(829)+PEEK(830)*256:GOTO7500                       <131>
7280 OPEN 2,8,2,PN$+",P,R"                                    <77>
7290 GET #2,A$:GET#2,B$                                       <125>
7300 CLOSE 2                                                  <164>
7310 EN=PEEK(174)+PEEK(175)*256                               <68>
7320 SS=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*
256                                                           <221>
7500 IF SS>SA+522 AND EN<=EA GOTO 7600                        <15>
7510 IF NOT((SS<SA AND EN<SA)OR SA>EA) G
OTO 7540                                                      <162>
7520 PRINT:PRINT"PRG{SPACE}LIEGT{SPACE}A
USSERHALB{SPACE}DER{SPACE}ERWEITERUNG":G
OTO 7570                                                      <20>
7540 IF SS>SA+522 AND EN>EA THEN PRINT:P
RINT"PRG{SPACE}IST{SPACE}ZU{SPACE}LANG!"
:GOTO 7470                                                    <220>
7550 PRINT:PRINT"BEREICH{SPACE}SA{SPACE}
BIS{SPACE}SA+522{SPACE}WURDE{SPACE}BESCH
RIEBEN"                                                       <212>
7570 PRINT:PRINT:PRINTTAB(10)"ZUM{SPACE}
MENUE{SPACE}MIT{SPACE RVSON}SPACE{RVSOFF
}"                                                            <113>
7580 GET A$:IF A$="{SPACE}" GOTO 3050                         <159>
7590 GOTO 7580                                                <41>
7600 PRINT:PRINT:INPUT"BEFEHLSWORT";BW$                       <46>
7610 IF SA+511-J<LEN(BW$)THEN PRINT:PRIN
T"ZU{SPACE}LANG":FOR T=1 TO 2000:NEXT:GO
TO 7600                                                       <18>
7700 PP$=LEFT$(BW$,LEN(BW$)-1)+CHR$(ASC(
RIGHT$(BW$,1))OR128)                                          <46>
7710 FOR I=1 TO LEN(PP$):POKE J+I-1,ASC(
MID$(PP$,I,1)):NEXT                                           <99>
7720 POKE SA+142+PEEK(SA+141)*2,FNL(SS)                       <225>
7730 POKE SA+143+PEEK(SA+141)*2,FNH(SS)                       <154>
7740 POKE SA+141,PEEK(SA+141)+1                               <161>
7750 PRINT:PRINT:PRINT"BEFEHL{SPACE RVSO
N}"BW$"{RVSOFF SPACE}IST{SPACE}EINGEBUND
EN."                                                          <48>
7760 PRINT:PRINT"STARTADRESSE:"SS                             <25>
7770 ZZ=1                                                     <96>
7780 PRINT:PRINT:PRINTTAB(10)"ZUM{SPACE}
MENUE{SPACE}MIT{SPACE RVSON}SPACE{RVSOFF
}"                                                            <68>
7790 GET A$:IF A$="{SPACE}" GOTO 3050                         <114>
7800 GOTO 7790                                                <6>
8000 REM                                                      <238>
8010 REM *****        ENDE        *****                       <83>
8020 REM                                                      <2>
8050 IF ZZ=0 GOTO 8150                                        <153>
8100 PRINT:PRINT:PRINT:PRINT                                  <11>
8110 PRINT"AENDERUNGEN{SPACE}SIND{SPACE}
NICHT{SPACE}GESPEICHERT!"                                     <99>
8120 PRINT"TROTZDEM{SPACE}PROGRAMM{SPACE
}BEENDEN{SPACE}(J/N)?"                                        <132>
8130 GET A$:IF A$="N" GOTO 3000                               <72>
8140 IF A$<>"J" GOTO 8130                                     <119>
8150 END                                                      <117>
9999 REM                                                      <197>
```

Wir hoffen, Ihnen eine nützliche Routine zur Befehlserweiterung vorgestellt zu haben und wünschen Ihnen viel Spaß bei der Anwendung.

Tips und Tricks für den C-64

Zwei nützliche Routinen: Verbesserter INPUT-Befehl und READ ERROR

Der verbesserte Input-Befehl

Bei den meisten Programmen, die den Input-Befehl verwenden, ist es sehr nützlich und nervenschonend, den Cursor und die Home Taste abzuschalten. Eine Möglichkeit ist die Verwendung von Pokes: **Poke 198,1:Poke 631,34** setzt ein Anführungszeichen in den Tastaturpuffer, Cursor und Home werden dann als inverse Zeichen dargestellt, das ist aber meistens auch nicht erwünscht.

Das folgende Maschinenprogramm bindet sich in die Tastatur-Dekodierroutine ein und unterdrückt den Cursor und die Hometaste vollkommen.

Mit SYS 710 wird Cursor und Home abgeschaltet, und nach dem Abschalten auch wieder aktiviert. SYS 710 sollte also vor und nach jedem Input-Befehl stehen.

```
50 FOR X=1 TO 53:READA:POKE 709+X,A:B=B+
A:NEXT:IF B=6212 THEN END                <218>
70 PRINT "DATA{SPACE}ERROR"              <50>
100 DATA 169,230,141,143,2,169,2,141,144
,2                                       <84>
110 DATA 173,229,2,73,1,141,229,2,208,10 <120>
120 DATA 169,72,141,143,2,169,235,141,14
4,2                                      <203>
130 DATA 96,0,165,203,201,2,240,8,201,7  <5>
140 DATA 240,4,201,51,208,4,169,64,133,2
03                                       <81>
150 DATA 76,72,235                       <185>
```

```
50 FOR X=1 TO 70:READ A:POKE 827+X,A:NEX
T:IF B=9101 THEN END                     <247>
100 DATA 169,71,141,143,2,169,3,141,144,
2                                        <8>
110 DATA 96,173,141,2,201,4,208,49,165,2
03                                       <72>
120 DATA 201,54,208,43,169,64,133,203,16
9,0                                      <108>
130 DATA 141,14,220,169,8,32,180,255,169
,111                                     <186>
140 DATA 133,185,32,150,255,169,0,133,14
4,32                                     <224>
150 DATA 165,255,32,210,255,165,144,240,
246,32                                   <185>
160 DATA 171,255,169,1,141,14,220,76,72,
235                                      <155>
```

READ ERROR

ist ein kurzes Maschinenspracheprogramm, das den Fehlerkanal der Floppy liest, wenn man CTRL und die Pfeil-nach-oben-Taste drückt. Durch

SYS 828

wird die Routine in die Tastaturdekodierung eingebunden und mit **RUN/STOP-Restore** wieder abgehängt.

Mit dem Commodore-64 im Meteoritensturm

Weltraumabenteuer mit tückischen Gefahren und enormen Flugbehinderungen

Sie sind in einem Raumschiff unterwegs, um neue Welten zu entdecken. Plötzlich geraten Sie in einen Meteoritensturm. Ihre Aufgabe ist es nun, das Raumschiff möglichst geschickt durch diese Gefahren zu manövrieren.

Stößt der Spieler an einen Meteoriten, ist das Spiel beendet. Je länger die Kapsel im Weltall unbeschadet bleibt, desto höher der Punktestand.

Gespielt wird mit einem Joystick (Port 2).

Achtung!

Hebel nach links bewirkt, daß die Kapsel sich nach rechts bewegt und umgekehrt. Oben und unten sind allerdings nicht vertauscht.

Nach dem Abtippen des Programms bitte erst absaven und dann normal mit **RUN** starten.

Bei einem **?DATA ERROR** wurden die DATA's falsch abgetippt. Sind die DATA's richtig, erscheint auf dem Bildschirm die **SYS-Adresse**, mit der das Programm gestartet wird, die dann nur noch mit der **RETURN-Taste** bestätigt werden muß. Es erscheint die Abfrage der Spielstärke, die mit den Tasten F1, F3 und F5 beantwortet wird.

Jeder weitere Tastendruck startet das Spiel.

```
100 REM ******************************   <105>
110 REM *                            *   <207>
120 REM *       METEORITENSTURM      *   <196>
130 REM *                            *   <227>
140 REM * COPYRIGHT '86 TRONIC-VERLAG *   <172>
150 REM *                            *   <247>
160 REM *   21.03.1986 SWEN OTTMANN  *   <225>
170 REM *                            *   <11>
180 REM *   $C000 BIS $C4C0 MASPRG.  *   <248>
190 REM *                            *   <31>
200 REM ******************************   <205>
210 :                                    <12>
220 DATA 3, 255, 0, 63, 255, 192, 255, 2
55, 240, 221                             <186>
230 DATA 254, 188, 213, 122, 172, 213, 1
22, 175, 213, 250                        <128>
240 DATA 171, 213, 234, 171, 213, 234, 1
71, 213, 234, 171                        <207>
250 DATA 245, 250, 171, 53, 126, 175, 53
, 94, 172, 61                            <201>
260 DATA 126, 172, 15, 254, 188, 3, 255,
 240, 3, 255                             <69>
270 DATA 192, 0, 255, 0, 0, 0, 0, 0, 0,
0                                        <42>
280 DATA 0, 0, 0, 0, 0, 0, 0, 1, 0, 0    <80>
290 DATA 5, 64, 0, 23, 84, 0, 23, 213, 6
4, 95                                    <226>
300 DATA 253, 80, 95, 175, 212, 94, 170,
 245, 94, 170                            <60>
310 DATA 245, 95, 175, 212, 95, 253, 80,
 23, 213, 64                             <106>
320 DATA 23, 84, 0, 5, 64, 0, 1, 0, 0, 0 <124>
330 DATA 0, 0, 0, 0, 0, 0, 0, 0, 0, 0    <114>
340 DATA 0, 0, 0, 0, 0, 0, 0, 0, 32, 68  <18>
350 DATA 229, 162, 14, 160, 11, 169, 1,
142, 32, 208                             <114>
360 DATA 140, 33, 208, 141, 134, 2, 169,
 0, 141, 30                              <247>
370 DATA 208, 169, 3, 141, 0, 16, 169, 4
8, 141, 0                                <32>
380 DATA 32, 141, 1, 32, 141, 2, 32, 141
, 3, 32                                  <126>
390 DATA 141, 4, 32, 141, 5, 32, 234, 23
4, 234, 160                              <215>
400 DATA 0, 185, 0, 192, 153, 64, 3, 200
, 192, 128                               <253>
410 DATA 208, 245, 162, 0, 160, 1, 169,
15, 142, 39                              <197>
420 DATA 208, 142, 40, 208, 140, 37, 208
, 141, 38, 208                           <202>
430 DATA 169, 3, 141, 21, 208, 162, 13,
160, 14, 142                             <58>
440 DATA 248, 7, 140, 249, 7, 162, 170,
160, 190, 142                            <106>
450 DATA 2, 208, 140, 3, 208, 169, 255,
141, 28, 208                             <101>
460 DATA 165, 203, 201, 64, 208, 3, 76,
240, 192, 32                             <119>
470 DATA 68, 229, 162, 60, 160, 140, 142
, 2, 208, 140                            <52>
480 DATA 3, 208, 234, 234, 234, 169, 13,
 141, 250, 7                             <141>
490 DATA 141, 251, 7, 141, 252, 7, 141,
253, 7, 141                              <109>
500 DATA 254, 7, 141, 255, 7, 169, 255,
141, 21, 208                             <24>
510 DATA 169, 0, 141, 41, 208, 141, 42,
208, 141, 43                             <246>
520 DATA 208, 141, 44, 208, 141, 45, 208
, 141, 46, 208                           <45>
530 DATA 162, 0, 160, 255, 169, 23, 142,
 134, 2, 140                             <214>
540 DATA 27, 208, 141, 17, 208, 169, 1,
133, 199, 162                            <39>
550 DATA 0, 160, 0, 185, 224, 195, 32, 2
10, 255, 200                             <247>
560 DATA 192, 40, 208, 245, 232, 224, 24
, 208, 238, 169                          <238>
570 DATA 0, 141, 32, 208, 32, 240, 193,
76, 130, 193                             <118>
580 DATA 72, 134, 251, 132, 252, 165, 25
3, 133, 2, 160                           <80>
590 DATA 0, 200, 192, 0, 208, 251, 198,
2, 208, 245                              <222>
600 DATA 166, 251, 164, 252, 104, 96, 20
6, 0, 208, 206                           <255>
610 DATA 4, 208, 206, 6, 208, 206, 8, 20
8, 206, 10                               <75>
620 DATA 208, 206, 12, 208, 206, 14, 208
, 32, 104, 193                           <235>
630 DATA 173, 0, 220, 41, 1, 201, 0, 240
, 30, 173                                <111>
640 DATA 0, 220, 41, 2, 201, 0, 240, 30,
 173, 0                                  <45>
650 DATA 220, 41, 4, 201, 0, 240, 30, 17
3, 0, 220                                <39>
660 DATA 41, 8, 201, 0, 240, 30, 76, 229
, 193, 206                               <95>
670 DATA 3, 208, 206, 3, 208, 76, 190, 1
93, 238, 3                               <15>
680 DATA 208, 238, 3, 208, 76, 190, 193,
 238, 2, 208                             <137>
690 DATA 238, 2, 208, 76, 190, 193, 206,
 2, 208, 206                             <187>
700 DATA 2, 208, 76, 190, 193, 32, 64, 1
94, 169, 0                               <88>
710 DATA 141, 30, 208, 76, 101, 193, 162
, 170, 160, 170                          <151>
720 DATA 142, 0, 208, 140, 1, 208, 162,
255, 160, 130                            <125>
730 DATA 142, 4, 208, 140, 5, 208, 162,
205, 160, 107                            <235>
740 DATA 142, 6, 208, 140, 7, 208, 162,
255, 160, 220                            <20>
750 DATA 142, 8, 208, 140, 9, 208, 162,
```

```
75, 160, 90                                                    <144>
760 DATA 142, 10, 208, 140, 11, 208, 162
, 99, 160, 190                                                 <84>
770 DATA 142, 12, 208, 140, 13, 208, 162
, 111, 160, 120                                                <13>
780 DATA 142, 14, 208, 140, 15, 208, 96,
 234, 234, 234                                                 <124>
790 DATA 234, 234, 234, 234, 234, 234, 1
20, 72, 173, 30                                                <129>
800 DATA 208, 201, 0, 240, 3, 76, 239, 1
94, 32, 86                                                     <205>
810 DATA 194, 104, 96, 104, 206, 32, 208
, 96, 238, 1                                                   <138>
820 DATA 208, 238, 5, 208, 238, 7, 208,
238, 9, 208                                                    <227>
830 DATA 238, 11, 208, 238, 13, 208, 238
, 15, 208, 169                                                 <177>
840 DATA 21, 141, 24, 212, 160, 255, 162
, 111, 140, 5                                                  <118>
850 DATA 212, 142, 6, 212, 162, 6, 160,
255, 142, 1                                                    <79>
860 DATA 212, 140, 4, 212, 169, 33, 141,
 4, 212, 238                                                   <181>
870 DATA 0, 32, 173, 0, 32, 201, 58, 240
, 1, 96                                                        <11>
880 DATA 169, 48, 141, 0, 32, 238, 1, 32
, 173, 1                                                       <73>
890 DATA 32, 201, 58, 240, 1, 96, 169, 4
8, 141, 1                                                      <208>
900 DATA 32, 238, 2, 32, 173, 2, 32, 201
, 58, 240                                                      <204>
910 DATA 1, 96, 169, 48, 141, 2, 32, 238
, 3, 32                                                        <50>
920 DATA 173, 3, 32, 201, 58, 240, 1, 96
, 169, 48                                                      <15>
930 DATA 141, 3, 32, 238, 4, 32, 173, 4,
 32, 201                                                       <85>
940 DATA 58, 240, 1, 96, 169, 48, 141, 0
, 32, 141                                                      <61>
950 DATA 1, 32, 141, 2, 32, 141, 3, 32,
141, 4                                                         <55>
960 DATA 32, 141, 5, 32, 96, 234, 234, 2
34, 234, 234                                                   <212>
970 DATA 234, 104, 169, 21, 141, 24, 212
, 160, 246, 162                                                <125>
980 DATA 8, 140, 5, 212, 142, 6, 212, 16
9, 48, 141                                                     <128>
990 DATA 1, 212, 169, 128, 141, 4, 212,
169, 129, 141                                                  <0>
1000 DATA 24, 212, 76, 16, 196, 234, 234
, 234, 234, 234                                                <112>
1010 DATA 234, 234, 234, 234, 234, 234,
234, 234, 234, 234                                             <138>
1020 DATA 32, 68, 229, 162, 6, 160, 12,
169, 0, 142                                                    <11>
1030 DATA 32, 208, 140, 33, 208, 141, 13
4, 2, 160, 0                                                   <30>
1040 DATA 185, 112, 195, 32, 210, 255, 2
00, 192, 103, 208                                              <237>
1050 DATA 245, 72, 169, 0, 141, 21, 208,
 141, 30, 208                                                  <225>
1060 DATA 104, 165, 203, 201, 4, 240, 11
, 201, 5, 240                                                  <94>
1070 DATA 14, 201, 6, 240, 15, 76, 73, 1
95, 169, 10                                                    <127>
1080 DATA 133, 253, 76, 96, 196, 169, 7,
 76, 92, 195                                                   <211>
1090 DATA 169, 4, 76, 92, 195, 234, 234,
 234, 234, 234                                                 <224>
1100 DATA 13, 13, 13, 32, 32, 32, 32, 32
, 32, 32                                                       <162>
1110 DATA 32, 32, 32, 32, 32, 77, 69, 84
, 69, 79                                                       <25>
1120 DATA 82, 73, 84, 69, 78, 83, 84, 85
, 82, 77                                                       <99>
1130 DATA 13, 13, 32, 32, 32, 32, 32, 32
, 32, 32                                                       <199>
1140 DATA 32, 32, 32, 49, 57, 56, 54, 32
, 83, 87                                                       <28>
1150 DATA 69, 78, 32, 79, 84, 84, 77, 65
, 78, 78                                                       <8>
1160 DATA 13, 13, 13, 70, 49, 32, 45, 32
, 65, 78                                                       <131>
1170 DATA 70, 65, 69, 78, 71, 69, 82, 13
, 70, 51                                                       <88>
1180 DATA 32, 45, 32, 70, 79, 82, 84, 71
, 69, 83                                                       <171>
1190 DATA 46, 13, 70, 53, 32, 45, 32, 80
, 82, 79                                                       <148>
1200 DATA 70, 73, 83, 73, 83, 234, 234,
234, 234, 234                                                  <7>
1210 DATA 234, 234, 29, 29, 29, 29, 29,
29, 29, 29                                                     <63>
1220 DATA 29, 29, 29, 29, 29, 29, 29, 29
, 29, 29                                                       <14>
1230 DATA 29, 29, 29, 29, 29, 29, 29, 29
, 29, 29                                                       <24>
1240 DATA 160, 160, 160, 160, 160, 160,
160, 160, 160, 160                                             <175>
1250 DATA 160, 160, 234, 234, 234, 234,
234, 234, 234, 234                                             <97>
1260 DATA 169, 0, 141, 21, 208, 32, 51,
196, 234, 234                                                  <235>
1270 DATA 234, 234, 234, 169, 128, 133,
253, 234, 234, 234                                             <21>
1280 DATA 234, 234, 160, 10, 32, 104, 19
3, 136, 208, 250                                               <186>
1290 DATA 76, 32, 195, 234, 234, 234, 17
3, 5, 32, 141                                                  <130>
1300 DATA 235, 5, 173, 4, 32, 141, 236,
5, 173, 3                                                      <154>
1310 DATA 32, 141, 237, 5, 173, 2, 32, 1
41, 238, 5                                                     <232>
1320 DATA 173, 1, 32, 141, 239, 5, 173,
0, 32, 141                                                     <52>
1330 DATA 240, 5, 96, 234, 234, 234, 234
, 234, 234, 255                                                <209>
1340 DATA 169, 0, 141, 21, 208, 162, 170
, 160, 170, 142                                                <52>
1350 DATA 0, 208, 140, 1, 208, 162, 255,
 160, 130, 142                                                 <254>
1360 DATA 4, 208, 140, 5, 208, 162, 205,
 160, 107, 142                                                 <80>
1370 DATA 6, 208, 140, 7, 208, 162, 255,
 160, 220, 142                                                 <101>
1380 DATA 8, 208, 140, 9, 208, 162, 75,
160, 90, 142                                                   <183>
1390 DATA 10, 208, 140, 11, 208, 162, 99
, 160, 190, 142                                                <160>
1400 DATA 12, 208, 140, 13, 208, 162, 11
1, 160, 120, 142                                               <163>
1410 DATA 14, 208, 140, 15, 208, 169, 0,
 141, 30, 208                                                  <144>
1420 DATA 141, 31, 208, 76, 128, 192, 0,
 255, 0, 255                                                   <177>
1430 DATA 0, 255, 0, 255, 0, 255, 0, 255
, 0, 999                                                       <173>
1440 :                                                         <223>
1450 CLR:X=49152:E=144827                                      <78>
1460 READY:IFY=999THEN1480                                     <150>
1470 POKEX,Y:P=P+Y:X=X+1:GOTO1460                              <1>
1480 IFP<>ETHENPRINT"?DATA{SPACE}ERROR":
END                                                            <169>
1490 PRINT"{DOWN}DATEN{SPACE}SIND{SPACE}
OK."                                                           <167>
1500 PRINT"{DOWN}START{SPACE}MIT{SPACE R
VSON}SYS49952{RVSOFF}"                                         <80>
1510 PRINT"{DOWN2}SYS49952{UP3}":END                           <119>
```

SA 2/86

Compute mit

COMMODORE

C=16

Heimcomputer

C=116/plus 4

ÖS 55 sfr 6,50 DM 6,50

SONDER-Ausgabe

50 mal können Sie diese Programm-Kassette gewinnen!

alle Programme mit Checksummen

Auch in dieser Ausgabe wieder interessante Beiträge zu den Themen:

- Werkstatt
- Tips & Tricks
- Software-Reviews
- Bücher
- Hardware
- u. v. m.

Sound-Machine – Key-Caller – Führung von Haushaltskonten – Textprogramm

Pac-Mac – Jäger des verlorenen Schatzes – Oil of Texas – Skateboarding

Mimi – Jäger der verlorenen Möhre – Soundhelfer – Auto-Starter – Ton-Clock

Neu: Sonderzeichen werden durch Klartext ersetzt!

Eingabe-Hinweise für Commodore C-16/116-Benutzer!

Um Ihnen die Eingabe unserer Listings zu erleichtern, werden zukünftig all C-16/116-Listings ohne Steuerzeichen abgedruckt. Alle Steuerzeichen werden in unseren LISTINGS durch Klartext, welcher die Taste kennzeichnet, ersetzt. Eine Tabelle der Tastenfunktionen finden Sie in jedem Tronic-Magazin.

Tastenbezeichnungen für Steuercodes (TRONIC-NORM)

DOWN	CURSOR UNTEN	TASTE NEBEN RECHTEM SHIFT
UP	CURSOR HOCH	SHIFT-TASTE & TASTE NEBEN RECHTEM SHIFT
CLR	CLEAR SCHIRM	SHIFT-TASTE & 2. TASTE GANZ RECHTS OBEN
INST	EINFUEGEN	SHIFT-TASTE & TASTE GANZ RECHTS OBEN
HOME	CURSOR IN ECKE	2. TASTE VON GANZ RECHTS OBEN
DEL	DELETE	TASTE GANZ RECHTS OBEN
RIGHT	CURSOR RECHTS	TASTE GANZ RECHTS UNTEN
LEFT	CURSOR LINKS	SHIFT-TASTE & TASTE UNTEN RECHTS
SPACE	LEERZEICHEN	LEERTASTE (GROESSTE TASTE)
F1	FUNKTIONSTASTE	FUNKTIONSTASTE F1
F3	FUNKTIONSTASTE	FUNKTIONSTASTE F3
F5	FUNKTIONSTASTE	FUNKTIONSTASTE F5
F7	FUNKTIONSTASTE	FUNKTIONSTASTE F7
F2	FUNKTIONSTASTE	FUNKTIONSTASTE F2
F4	FUNKTIONSTASTE	FUNKTIONSTASTE F4
F6	FUNKTIONSTASTE	FUNKTIONSTASTE F6
F8	FUNKTIONSTASTE	FUNKTIONSTASTE F8
BLACK	SCHWARZ	CONTROL-TASTE & 1
WHITE	WEISS	CONTROL-TASTE & 2
RED	ROT	CONTROL-TASTE & 3
CYAN	TUERKIS	CONTROL-TASTE & 4
PURPLE	PURPUR	CONTROL-TASTE & 5
GREEN	GRUEN	CONTROL-TASTE & 6
BLUE	BLAU	CONTROL-TASTE & 7
YELLOW	GELB	CONTROL-TASTE & 8
RVSON	INVERSE EIN	CONTROL-TASTE & 9
RVSOFF	INVERSE AUS	CONTROL-TASTE & 0
ORANGE	ORANGE	COMMODORE-TASTE & 1
BROWN	BRAUN	COMMODORE-TASTE & 2
LIG.RED	HELLROT	COMMODORE-TASTE & 3
DGREY	DUNKELGRAU	COMMODORE-TASTE & 4
MGREY	MITTELGRAU	COMMODORE-TASTE & 5
LIG.GREEN	HELLGRUEN	COMMODORE-TASTE & 6
LIG.BLUE	HELLBLAU	COMMODORE-TASTE & 7
HGREY	HELLGRAU	COMMODORE-TASTE & 8
CTRL	CONTROL	CONTROL-TASTE ZUSAMMEN MIT DEM NACHFOLGENDEN ZEICHEN. (Z.B. CTRLA ENTSPRICHT CTRL & A)
FLASH ON	BLINKEN EIN	CONTROL-TASTE &,
FLASH OFF	BLINKEN AUS	CONTROL-TASTE &.

Achtung: Einige Listings enthalten das Zeichen "–", dieses Zeichen entspricht dem Pfeil nach links (Taste ganz links oben)
das Zeichen " ∧ " steht für den Pfeil nach oben, (SHIFT/Null)
das Zeichen "\" steht für das (engl.) Pfund-Symbol (Taste neben Restore)

Folgendes Beispiel demonstriert die Arbeitsweise unseres Systems:

```
STANDARD AUSDRUCK:

10 A$="TEST":PRINT"[redacted]HALLO[redacted]LEUTE !":PRINT"TRONIC VERLAG"
20 PRINT"[redacted]ENDE"
```

```
UNSER NEUER AUSDRUCK DER GLEICHEN ZEILEN:

10 A$="TEST":PRINT"{CLEAR RIGHT3 DOWN3 R
ED}HALLO{RIGHT2 BLUE}LEUTE{SPACE}!":PRIN
T"TRONIC{SPACE}VERLAG"                <34>
20 PRINT"{CYAN PURPLE GREEN}ENDE"     <117>
```

Achtung !!!!

Kurz vor Redaktionsschluß bemerkten wir, daß die Klartext-Tabelle mit den Schlüsselworten beim C-16/116 nicht ganz zutrifft. In erster Linie sind hier die Farben betroffen. Zwar stimmt die Tastenbelegung genau mit den abgedruckten Code's überein, jedoch erscheint bei einigen Tastenkombinationen eine andere Farbe auf dem Bildschirm. Die nachfolgende Gegenüberstellung klärt hier auf:

Abgedruckter Klartext	Richtige Farbe
LIG.RED	Gelb-Grün
DGREY	Rosa
MGREY	Blau-Grün
LIG.GREEN	Hellblau
LIG.BLUE	Dunkelblau
HGREY	Hellgrün

Ebenfalls stimmen einige Tastenbeschreibungen nicht ganz überein, da der C-16/116 die Cursortasten sowie CLR und DEL anders angeordnet hat. Da diese Tasten jedoch beschriftet sind, dürfte es hier keine Probleme geben. In zukünftigen Ausgaben werden wir eine vollständig neue Tabelle abdrucken.

Wie in dem Beispiel zu erkennen ist, wird das Herz-Symbol durch das Wort CLEAR ersetzt. Dies bedeutet, Sie müssen die Taste CLEAR drücken, um das Herz-Symbol zu erzeugen. Sollten Sie einmal nicht wissen, welche Taste gemeint ist, hilft Ihnen ein Blick in unsere Tastentabelle bestimmt weiter.

Um im Listing Tastenbezeichnungen von Grafik oder normalen Texten zu unterscheiden, werden alle Tastenkennzeichnungen in geschweifte Klammern gesetzt. Diese dürfen selbstverständlich nicht eingegeben werden. Auch Leerzeichen innerhalb geschweifter Klammern dienen nur zur Trennung einzelner Tastenfunktionen und dürfen ebenfalls nicht eingegeben werden. Um die Tastenfunktionen noch besser hervorzuheben, werden diese in unterstrichener Kursivschrift (Schrägschrift) abgedruckt.

Steht hinter einer Tastenfunktion eine Zahl, welche ebenfalls unterstrichen ist, bedeutet dies, daß die letzte Tastenfunktion mehrmals betätigt werden muß. Die in unserem Beispiel abgedruckte Funktion RIGHT3 bedeutet, daß die Taste RIGHT (Cursor rechts) 3 mal hintereinander betätigt werden muß. Auch einzelne oder mehrere Leerzeichen innerhalb von Anführungszeichen werden auf diese Art gekennzeichnet. Das bislang übliche Abzählen einzelner Zeichen entfällt somit völlig.

Alle Zeichen außerhalb von den geschweiften Klammern werden normal abgedruckt und auch eingegeben.

Auf den ersten Blick hört sich das sicher etwas kompliziert an, ist jedoch in der Praxis ganz einfach. Wenn man sich erst an die in Klartext geschriebenen Steuerzeichen gewöhnt hat, wird man den großen Vorteil dieser Schreibweise erkennen.

Checksummer OC Version 1.0 für C-16/116

Die Arbeitsweise des C-16/116 Checksummers entspricht weitgehend der VC-20- und C-64-Version und braucht deshalb nicht näher erklärt zu werden. Da auch der C-16/116 nicht den größten Speicher besitzt, wurde hier die Methode der VC-20-Version gewählt und das Maschinenprogramm in den Kassettenpuffer gelegt. Dies hat den Vorteil, daß kein Basic-Speicher verlorengeht. Der Nachteil besteht darin, daß nach der Aktivierung des Checksummers keine Kassettenoperationen durchgeführt werden dürfen, da diese unweigerlich zum Systemabsturz führen. Um die Kassettenoperationen wieder zuzulassen, muß der Checksummer durch die beiden folgenden SYS-Aufrufe abgeschaltet werden:

SYS 62158 : SYS 33047

Aktiviert wird der Checksummer mit **SYS 818**. Die Funktion und Arbeitsweise entnehmen Sie bitte der C64-Checksummer-Version. Eine Ausnahme bei C-16/116 sind die zusätzlichen Steuerzeichen **'FLASHON' (CTRL-,)** und **'FLASHOFF' (CTRL.)** welche ebenfalls durch Klartext ersetzt werden (s. Tabelle Tastenbezeichnungen).

Arbeitsweise und Aufbau unseres Checksummers:

Unser Checksummer besteht aus einem kleinen Maschinenprogramm, welches als Basic-Loader abgedruckt ist.

Tippen Sie diesen Loader ein und speichern Ihn auf Kassette oder Diskette, denn Sie können ihn zukünftig immer wieder benutzen.

Der Start erfolgt durch den Befehl „RUN". Nach kurzer Zeit meldet sich der Rechner mit der Meldung „TRONIC ...". Der Checksummer ist nun aktiv und man kann ein beliebiges Tronic-Listing eingeben. Nachdem eine Zeile mit RETURN abgeschlossen wird, erscheint links oben auf dem Schirm eine Prüfzahl. Vergleichen Sie diese mit der Zahl im Heft hinter der Zeile. Stimmt die Zahl überein, ist die Zeile richtig eingegeben.

Auf diese Weise können Sie das gesamte Listing schnell und fehlerfrei eingeben.

Interessant ist auch, daß bei der Eingabe von Zeilen die üblichen Abkürzungen benutzt werden können, ohne die Checksumme zu verändern. Leerzeichen außerhalb von Anführungszeichen werden ignoriert, da diese auf die Ausführung der einzelnen Befehle keinen Einfluß haben.

Für alle, die nicht gerne abtippen, ist der Checksummer unter folgender Bestellnummer zu beziehen:

OV 10 K Kassette 10 DM
OV 10 D Diskette 15 DM

```
1 REM ********************************  <4>
2 REM *                              *  <99>
3 REM *   C16-CHECKSUMMER             *  <140>
4 REM *                              *  <101>
5 REM *   COPYRIGHT BY FRANK BRALL    *  <182>
6 REM *                              *  <103>
7 REM *   (C) 18.04.86                *  <181>
8 REM *                              *  <105>
9 REM ********************************  <12>
10 PRINT"{CLEAR DOWN SPACE2}◆◆◆◆{SPACE2}
CHECKSUMMER{SPACE}OC{SPACE}1.0{SPACE2}◆◆
◆◆{DOWN}"                                <103>
11 PRINT"{SPACE2}◆◆{SPACE2}COPYRIGHT{SPA
CE}FRANK{SPACE}BRALL{SPACE3}◆◆{DOWN}"    <110>
12 PRINT"{SPACE8}FUER{SPACE}TRONIC-VERLA
G"                                       <196>
60000 DIMH(75):FORI=0TO9                 <123>
60010 H(48+I)=I:H(65+I)=I+10:NEXT        <145>
60020 FORI= 818 TO 1010 :READA$          <151>
60030 H=ASC(LEFT$(A$,1))                 <234>
60040 L=ASC(RIGHT$(A$,1))                <253>
60050 D=H(H)*16+H(L):S=S+D:POKEI,D       <13>
60060 A=A+1:IFA<20THENNEXT:A=-1          <97>
60070 READV:Z=Z+1:IFV=STHEN60085         <177>
60080 PRINT"DATAFEHLER{SPACE}IN{SPACE}ZE
ILE{SPACE}:";60200+Z:END                 <39>
60085 IFA<0THEN60100                     <116>
60090 S=0:A=0:NEXT                       <246>
60100 PRINT"{DOWN2}"                     <172>
60110 PRINT"CHECKSUMMER{SPACE}EIN{SPACE}
={SPACE}SYS{SPACE}818{DOWN}"             <140>
60120 PRINT"CHECKSUMMER{SPACE}AUS{SPACE}
={SPACE}SYS{SPACE}62158+SYS{SPACE}33047{
DOWN}"                                   <122>
60130 PRINT"{DOWN}SAVE/LOAD{SPACE}IST{SP
ACE}WAEHREND{SPACE}DER{SPACE}CHECK-"     <61>
60140 PRINT"SUMMER{SPACE}AKTIV{SPACE}IST
,{SPACE}NICHT{SPACE}MOEGLICH{SPACE}!"    <221>
60150 PRINT"{DOWN}BEACHTEN{SPACE}SIE{SPA
CE}DIE{SPACE}HINWEISE{SPACE}IN{SPACE}DEN
"                                        <189>
60170 PRINT"HEFTEN{SPACE}COMPUTE{SPACE}M
IT{SPACE}UND{SPACE}COMPUTRONIC"          <234>
60180 PRINT"{DOWN}VIEL{SPACE}SPASS{SPACE
}!{SPACE}(AUTOR:F.BRALL/6443SONTRA)      <116>
60190 POKE 814,147:POKE815,251:POKE 816,
147:POKE 817,251                         <109>
60195 SYS 818:END                        <177>
60201 DATA A9,3D,8D,02,03,A9,03,8D,03,03
,60,A2,FF,86,3A,20,5A,88,86,3B, 1851     <56>
60202 DATA 84,3C,20,73,04,AA,F0,EF,90,09
,20,53,89,20,79,04,4C,D9,88,20, 2018     <28>
60203 DATA 3E,8E,20,53,89,84,0B,A9,00,8D
,E4,03,8D,E5,03,8D,E6,03,18,A5, 2076     <40>
60204 DATA 14,65,15,8D,E6,03,A0,FF,C8,B1
,3B,F0,2C,C9,22,D0,0A,AD,E4,03, 2508     <18>
60205 DATA 49,01,8D,E4,03,B1,3B,AE,E4,03
,D0,04,C9,20,F0,E4,EE,E5,03,AE, 2644     <87>
60206 DATA E5,03,18,B1,3B,6D,E6,03,8D,E6
,03,CA,D0,F4,4C,76,03,3B,20,39, 2204     <107>
60207 DATA D8,8E,E2,03,8C,E3,03,A2,00,A0
,00,18,20,39,D8,A9,5B,20,D2,FF, 2365     <115>
60208 DATA A9,00,AE,E6,03,20,5F,A4,A9,5D
,20,D2,FF,20,08,FB,20,08,FB,AC, 2380     <169>
60209 DATA E3,03,AE,E2,03,18,20,39,D8,A2
,00,86,FF,4C,36,87,00,00,00,00, 1778     <133>
60210 DATA 00,00,00,00,00,00,00,00,00,00
,00,00,00, 0                             <104>
```

Zeitschriften und Programme von 1985/86

Heft 5/85

Commodore
Bestell-Nr. COM CK 1/1 16,– DM
Bestell-Nr. COM CD 1/1 20,– DM
Senso (C-16/116), Jungle (C-64), Autostart (C-64), Sequenzer (C-64), Safe (C-64), Olympia (VC-20), Flowers (VC-20)

Schneider
Bestell-Nr. COM SK 1/1 16,– DM

Cobra, Krümi, Haunted Castle, Q-Bert, ROM-CALLS

Heft 6/85

Commodore
Bestell-Nr. COM CK 2/1 16,– DM
Bestell-Nr. COM CD 2/1 20,– DM
Mampfman (C-64), Softlanding (C-64), Karate (C-64), Einbrecher (VC-20), Tanker (VC-20), Jupiter Rescue (VC-20), SOS-Schiffbruch (C-16)

Schneider
Bestell-Nr. COM SK 2/1 16,– DM

Tastaturbelegung, Cave-Runner, Old-Shurehand

Heft 7/85

Commodore
Bestell-Nr. COM CK 3/1 16,– DM
Bestell-Nr. COM CD 3/1 20,– DM
Sea-Attak (C-64), Sace-Ape (C-64), Micro-Tools (C-64), Disky 1.0 (C-64), Stuntman (VC-20), Sepp (VC-20), Balloon-Shoot (VC-20), Farbenfroh (C-16)

Schneider
Bestell-Nr. COM SK 3/1 16,– DM
Bestell-Nr. COM SD 3/1 39,– DM
Secret Valley, Smileymat, Super-File

Heft 8/85

Commodore
Bestell-Nr. COM CK 4/1 16,– DM
Bestell-Nr. COM CD 4/1 20,– DM
Bracker-List (VC-20), Flowers II (VC-20), Dino Eggs (VC-20), Piano (C-16/116), Tank-Duell (C64), Vampir (C64), Memory (C-64), Raster Register, Interrupt (C-64).

Schneider
Bestell-Nr. COM SK 4/1 16,– DM
Bestell-Nr. COM SD 4/1 39,– DM
Copter Patrol, Suchrätsel, Tower

Heft 9/85

Commodore
Bestell-Nr. COM CK 5/1 16,– DM
Bestell-Nr. COM CD 5/1 20,– DM
Supermoni 64 (C-64), Galaxy (C-64), Atlantis (C-64), Shaolin (C-16), Hospital (VC-20), Flipper-Ball (VC-20), Alien (VC-20)

Schneider
Bestell-Nr. COM SK 5/1 16,– DM
Bestell-Nr. COM SD 5/1 39,– DM
Duell, Crash, Citylander, Klondike, REM-Killer

Heft 10/85

Commodore
Bestell-Nr. COM CK 6/1 16,– DM
Bestell-Nr. COM CD 6/1 20,– DM
The Castle of Dr. Creep (VC-20), Catchge (VC-20), Sub-Zone (VC-20), The Maze Graze Game (C-64), Super-Breakout (C-64), Planet of Death (C-64), Funktionstasten-Programmierung (C-64), Car Race (C-16/116), Key-Helper (C-16/116)

Schneider
Bestell-Nr. COM SK 6/1 16,– DM
Bestell-Nr. COM SD 6/1 39,– DM
Labyrinth, Catch, Mini Car Race, Inka Sogra, Single Disk Backup

Heft 11/85

Commodore
Bestell-Nr. COM CK 7/1 16,– DM
Bestell-Nr. COM CD 7/1 20,– DM
Interpreter Music-Language (C-64), Firelegion (C-64), Funktionstastenprogrammierung (C-64), Schatztaucher (C-16), Dangerous Planet (VC-20), Gee-Zaxxon (VC-20)

Schneider
Bestell-Nr. COM SK 7/1 16,– DM
Bestell-Nr. COM SD 7/1 39,– DM
Etis, Alien-Hunter, Demon-Attack, Datenverwaltung, Discshow

Heft 12/85

Commodore
Bestell-Nr. COM CK 4/1 16,– DM
Bestell-Nr. COM CD 4/1 20,– DM
Froggies Nightmare (VC-20), Wächter des heiligen Tempels (VC-20), Jumping Joe (C-16/116), City-Runner (C-16/116), Mission Rescue (C-64), Turbo-Jet (C-64), Print-Generator (C-64), Datagenerator (C-64)

Schneider
Bestell-Nr. COM SK 8/1 16,– DM
Bestell-Nr. COM SD 8/1 39,– DM
Checksummer 1.0 CPC, Two Cherry Chaps, Erwin im Zauberschloß, Drucksteuerung NLQ 401

Heft 1/86

Commodore
Bestell-Nr. COM CK 1/2 16,– DM
Bestell-Nr. COM CD 1/2 20,– DM
Checksummer 1.0 C, Astronaut Battle (C-64), Hueycobra (C-16/116), Hardcopyroutine (C-16/116), Helmut und die Zeitmaschine (C-16/116), Locos (VC-20), Rock'me (VC-20), Charly der Raketenmann (VC-20), Interrupt-Programmierung (VC-20)

Schneider
Bestell-Nr. COM SK 1/2 16,– DM
Bestell-Nr. COM SD 1/2 39,– DM
Defender, Boulder Dash, Basic-Befehle auf den Funktionstasten

Heft 2/86

Commodore
Bestell-Nr. COM CK 2/2 16,– DM
Bestell-Nr. COM CD 2/2 20,– DM
Robo (C-64), Basic-Befehlserweiterung (C-64), Raketenstart (C-16/116), Dreikampf (C-16/116), Headline (C-16/116), Defender (VC-20), Formel 1 (VC-20), Memory-Display (VC-20)

Schneider
Bestell-Nr. COM SK 2/2 16,– DM
Bestell-Nr. COM SD 2/2 39,– DM
Alien 2001, Diamond Hunter, Directory

Heft 3/86

Commodore
Bestell-Nr. COM CK 3/2 16,– DM
Bestell-Nr. COM CD 3/2 20,– DM
Minor-Jumper (C-16/116), Nero (C-16/116), Data-Maker (C-16/116), Befehlserweiterung (C-64), Starchip (C-64), Mini-Car-Race (C-64), Text-Scroller (C-64), Text-Editor (C-64), Checksummer (VC-20), Robin at Ravenwreck (VC-20), Mini-Karate (VC-20), Autonumber-Routine (VC-20)

Schneider
Bestell-Nr. COM SK 3/2 16,– DM
Bestell-Nr. COM SD 3/2 39,– DM
Smily, Screpi, Die verrückte Fabrik, Disc-Help

Heft 4/86

Commodore
Bestell-Nr. COM CK 4/2 16,– DM
Bestell-Nr. COM CD 4/2 20,– DM
Shuttle (VC-20), Josef in den Höhlen (VC-20), Merge-Routine (VC-20), Shoot it (VC-20), Fast Trust (C-64), Data-Text 64 (C-64), Gary, der schnelle Kabelverleger (C-16/116), SDI (C-16/116), Beep-Routine (C-16/116)

Schneider
Bestell-Nr. COM SK 4/2 16,– DM
Bestell-Nr. COM SD 4/2 39,– DM
Space-Rescue, Texteditor, Starfight, Programm-Retter

Heft 5/86

Commodore
Bestell-Nr. COM CK 5/2 16,– DM
Bestell-Nr. COM CD 5/2 20,– DM
Space-Devil (C-64), Disk-Eddi (C-64), ABC-Legen (V-20), 3-D-Labyrinth (VC-20), Tron (VC-20), Invaders of Space (C-16/116), Submarine (C-16/116), Balkondiagramm (C-16/116)

Schneider
Bestell-Nr. COM SK 5/2 16,– DM
Bestell-Nr. COM SD 5/2 39,– DM
Startank, Grafik-Designer, Cross-Reference, Itext

Heft 6/86

Commodore
Bestell-Nr. COM CK 6/2 16,– DM
Bestell-Nr. COM CD 6/2 20,– DM
Music-Master (C-64), The little Puck goes home (C-64), EDDY (C-64), Invers-Utility (C-16/116), Checksummer (C-16/116), Affenwurf (C-16/116), Data-Maker (VC-20), Karate-Kid (VC-20), Q-Bert (VC-20)

Schneider
Bestell-Nr. COM SK 6/2 16,– DM
Bestell-Nr. COM SD 6/2 39,– DM
Disksort, Mini-Forth, Starcommand

Aus diesem Heft

Commodore
Bestell-Nr. COM CK 7/2 16,– DM
Bestell-Nr. COM CD 7/2 20,– DM

Schneider
Bestell-Nr. COM SK 7/2 16,– DM
Bestell-Nr. COM SD 7/2 39,– DM

Telefonische Bestellungen unter
05651/30011
Bearbeitung innerhalb einer Woche

Alle hier aufgeführten Kassetten und Disketten können jederzeit nachbestellt werden. Bei den Zeitschriften sind die Ausgaben 6/85, 7/85, 11/85, 2/86 und 3/86 leider vergriffen und können nicht mehr nachgeliefert werden. Bitte beachten Sie dies bei Ihrer Bestellung.

C=64

Programm	Preis/ Kassette	Preis/ Diskette	Bestell-Nr.
Multi-Key/S-Tool 64/ Interrupt-Programme	16,-- DM	21,-- DM	UC 2/85
Spritehilfe/Diskloader/ Directory	21,-- DM	-	UC 3.1/85
Tape-Directory/Asmon/ Data-Generator	15,-- DM	21,-- DM	UC 3.2/85
Fast Load	10,-- DM		UC 4/84
Diskmonitor/Zeichensatz	20,-- DM	-	UC 5/85
Reassembler/Maskengenerator	15,-- DM	20,-- DM	UC 6/85
Del 64/Treiberprogramm für 1526/MPS 802 Decelevator	14,-- DM	19,-- DM	UC 7/85
Supertapedirectory/Renew/ Kontrollabfrage	15,-- DM	-	UC 8/85
Video-Utilities, Multi-Basic V3, Bildeditor	17,-- DM	23,-- DM	UC 9/85
Better Basic/Autonumber/ Hardcopy/Terminalprogramm	17,-- DM	23,-- DM	UC 10/85
Speicher-Oszilloskop/ Makro-Assembler/Change Type	49,-- DM	54,-- DM	UC 1/86
Reactor/Concentration/Datenbank	17,50 DM	23,50 DM	HC/C-1
Warlords/Caverns of Death	14,50 DM	19,50 DM	HC/C-2
Alien-Destroyer/Duell	24,50 DM	29,50 DM	HC/C-3
Ocean Game/Tennis	17,50 DM	23,50 DM	HC/C-4
The Caves	17,50 DM	23,50 DM	HC/C-5
Gardener	17,50 DM	23,50 DM	HC/C-6
Ufo/Skateboard Sam	24,50 DM	29,50 DM	HC/C-1-4
Olympic Biathlon/ Struggle for Life	24,50 DM	29,50 DM	HC/C-2-4
Pole Position/Save your teeth/Push	24,50 DM	29,50	HC/C-3-4
Explorer 2/Programmname?	24,50 DM	29,50 DM	HC/C-4-4
Mauern/Widerstand	8,-- DM	15,-- DM	C 41
Space-Comets/Erdspalte/ Sprite-Data	15,-- DM	23,50 DM	C 51
Autostart/Bestellschein/ Roadpainter	16,50 DM	23,50 DM	C 61
Hardcopy/Space-Fighter/ Data-Generator	15,50 DM	19,50 DM	C 71
Monster-Attack/Block-Painter/Epson-Drucker	16,50 DM	23,50 DM	C 81
Projekt/Datenbank	16,-- DM	23,50 DM	C 91
Spiders/The Basic	16,50 DM	23,50 DM	C 101
High Noon/Skeet/ Grafik-Designer	17,50 DM	23,50 DM	C 121
Painter/Star-Baddle/Editor	17,50 DM	23,50 DM	C 22
Wüstenrally/Jet-Pac/ Black Moore Castle	17,50 DM	23,50 DM	C 32
Brieftaube/Cadelon	19,50 DM	24,50 DM	C 42
Ritter Erik/Grand Prix Spritehilfe	19,50 DM	24,50 DM	C 52
Firebird/High Music	24,50 DM	29,50 DM	C 62
Moonsweeper/Scotti	24,50 DM	29,50 DM	C 72
Tron/Mercurious	24,50 DM	29,50 DM	C 13
Interceptor-Basel/Schotter	24,50 DM	29,50 DM	C 23
Jump-Man/Space-Taxi/The Way	24,50 DM	29,50 DM	C 33

Neu in unserer Software-Bestell-liste der

C=16

Fantasy-Country/ Horror-Castle	18,-- DM	23,-- DM	O 23
Moon Fighter/Goldrausch	18,-- DM	23,-- DM	O 33
Zeichendesigner	14,-- DM	-	UO 9/85
OLD-Routine-Merge-Routine/ Change Type	14,--	19,--	UO 1/86

VIC-20

Programm	Preis/ Kassette	Preis/ Diskette	Bestell-Nr.
Bestellschein/Glücky	8,50 DM	15,- DM	V 61
Multigraph/All Rammer	11,-- DM	15,50 DM	V 71
Zyklo/Meteorit	11,-- DM	19,50 DM	V 81
Garten/Schloß Gruselstein	14,-- DM	19,50 DM	V 91
Fressman/Outlaw	14,-- DM	19,50 DM	V 101
Prost/Buffalo Bill	14,-- DM	19,50 DM	V 121
Joy Man/Powerpack	14,-- DM	19,50 DM	V 22
Der rasende Maloocher/ Frankie goes to Pharao	14,-- DM	19,50 DM	V 32
Matron/Obst	14,-- DM	19,50 DM	V 42
Race On/Copy	14,-- DM	19,50 DM	V 52
Nürburg 3D/Düsi	14,-- DM	19,50 DM	V 62
Breaker/Expulsion	14,-- DM	19,50 DM	V 72
Brückenbau/Jange	16,-- DM	21,50 DM	V 13
Inka-Schatz/ Miner der Fassadenstreicher	16,-- DM	21,50 DM	V 23
Booby In Action/Bergshooting	16,-- DM	21,50 DM	V 33
Zyklo/Geldsammler	11,50 DM	-	HC/V-1
Star-Wars/Punktefresser	14,50 DM	-	HC/V-2
Catch the Fish/Mister Jump	14,50 DM	-	HC/V-3
Ghost's Eggs/Fox Hunt	14,50 DM	-	HC/V-4
Collector/Break Out	14,50 DM	-	HC/V-5
Q-Bert/Salvage Crew	16,50 DM	-	HC/V-6
Dog Fight/Schwitzbad	16,50 DM	-	HC/V-1-4
Earth Rescue	14,50 DM	-	HC/V-2-4
Tank	14,50 DM	-	HC-V-3-4
Terra 2/Pac-Ball	16,50 DM	21,50 DM	HC/V-4-4
Sprites/Space-Battle	13,-- DM	19,-- DM	UV 2/85
Grafik-Painter	10,-- DM	-	UV 5/85
VIC-Clock	9,-- DM	-	UV 6/85
Decelerator	9,-- DM	-	UV 7/85
Joypaint	12,-- DM	-	UV 8/85
Disassembler, Old-Programme	10,-- DM	-	9/85
Directory/Farb-JRQ	10,-- DM	-	UV 10/85

Schneider CPC 464

Autorennen	11,-- DM	-	US 3/85
Universal-Datei V1	9,-- DM	-	US 4/85
Computerschrift/Symbol-Swap	10,-- DM	-	US 5/85
Keyboard Toolkit/Farbdecodierung	11,-- DM	-	US 6.1/85
Aladin	9,-- DM	-	US 6.2/85
Painter/Box-Befehl	14,-- DM	24,-- DM	US 7/85
Maschinensprache-Monitor/ Disk-Hilfe	15,-- DM	25,-- DM	US 8/85
Basic, Maschinen-Kit	12,-- DM	22,-- DM	US 9/85
Backup/Directory/ Ellipse/Disk	-	27,-- DM	US 10/85
Data-Generator/Grafik Toolkit/ ASCII List (Turbo Pascal)	15,-- DM	25,-- DM	US 1/86
Fallschirmspringer	9,-- DM	-	HC/SR-3
Geisterschloß	9,-- DM	-	HC/SR-4
Zeichendesigner	12,50 DM	-	HC/SR-5
Mini Car Race/Interceptor 3 D	17,50 DM	26,50 DM	HC/SR-6
Secret Agent	12,50	24,50	HC/SR-1-4
Space Invaders/Mini Pac/ Ottl's „Sound"	17,50 DM	27,50 DM	HC/SR-2-4
Samson und die Kisten/ Die verflixte 15	17,50 DM	27,50 DM	HC/SR-3-4
Moon-Landing/Panzer	17,50 DM	26,50 DM	HC/SR-4-4
Super-Miner	14,-- DM	24,-- DM	SR 42
CPC-Bert	14,-- DM	24,-- DM	SR 52
Conan's Castle	15,50 DM	25,50 DM	SR 62
Snider's Maca	16,50	26,50 DM	SR 72
Schneider Panik/ Killership	18,50 DM	28,50 DM	SR 13
Horror-Caves/ Midnight	18,50 DM	28,50 DM	SR 23
Cosmis-Castle/Spider-Mace/ Traumland	18,50 DM	28,50 DM	SR 33

Hubschrauberaction

für den C-16/116

Airwolf

Im Auftrag ihrer Regierung müssen Sie versuchen als Pilot des Airwolfs einen alten Kriegsschatz zu bergen. Dies ist aber so nicht möglich, da eine Bombe den Eingang des Gewölbes, in der der Schatz verborgen ist, versperrt. Sie müssen nun versuchen, in diese Höhle hineinzufliegen und die Bombe mit Hilfe Ihrer eigenen Bomben zur Explosion zu bringen. Wenn Sie das geschafft haben, haben Sie nur 50 Zeiteinheiten, um aus der Höhle wieder herauszukommen und an ihren Startpunkt zu fliegen. Dies ist aber nicht so einfach wie es klingt. Zusätzlich behindern Sie Minen, die noch vom 2. Weltkrieg stammen und plötzlich herunterfallende Felsbrocken, die Ihren Hubschrauber gefährden.

Die Steuerung erfolgt per Joystick in Port 1.

Eingabehinweise:
Das Programm besteht aus zwei Teilprogrammen:
1. Airwolf-Datas
2. Airwolf-Spiel

Die Programme müssen nacheinander getippt, gespeichert, geladen und gestartet werden.

Hinweise zum Abwerfen der Bomben und Raketen:
Joystickfeuerknopf = Bombe wird abgeworfen
Joystickfeuerknopf + Ausschlag des Joysticks = Rakete wird in die Flugrichtung abgeschossen.

Mine = 30 Punkte
Felsbrocken = 150 Punkte
Bombe = 300 Punkte
Erreichen des Startpunktes nach Abschießen der Bombe = 1000 Punkte

```
100 REM ********************************  <211>
110 REM *      --- AIRWOLF-DATAS ---     *  <164>
120 REM ********************************  <231>
130 REM *          WRITTEN BY            *  <168>
140 REM **                              **  <103>
150 REM ***    ANDREAS POLEFKA         ***  <103>
160 REM ****                          ****  <103>
170 REM ***    5470 ANDERNACH 1        ***  <241>
180 REM **                              **  <143>
190 REM *          [C] 1986              *  <142>
200 REM ********************************  <55>
210 COLOR0,1                               <97>
220 COLOR4,3,3                             <137>
230 COLOR1,2                               <123>
240 POKE0,0                                <43>
250 PRINT"{CLEAR DOWN RVSON SPACE13}AIRW
OLF{SPACE}DATAS{SPACE14 RVSOFF}"           <9>
260 PRINT"{DOWN2 RIGHT}BITTE{SPACE}DRUEC
KEN{SPACE}SIE{SPACE}DIE{SPACE RVSON}PLAY
{RVSOFF SPACE}TASTE."                      <72>
270 PRINT"{DOWN RIGHT}DAMIT{SPACE}DER{SP
ACE}COMPUTER{SPACE}NACH{SPACE}DEN{SPACE}
EINLESEN"                                  <80>
280 PRINT"{DOWN RIGHT}AUTOMATISCH{SPACE}
DEN{SPACE}LADEVORGANG{SPACE}FUER{SPACE}D
AS"                                        <82>
290 PRINT"{DOWN RIGHT}SPIEL{SPACE}VORNEH
MEN{SPACE}KANN."                           <218>
300 PRINT"{DOWN RIGHT}WENN{SPACE}SIE{SPA
CE}DIE{SPACE}DATAS{SPACE}JETZT{SPACE}EIN
LESEN"                                     <103>
310 PRINT"{DOWN RIGHT}WOLLEN{SPACE}DANN{
SPACE}DRUECKEN{SPACE}SIE{SPACE}DIE{SPACE
 RVSON}RETURN{RVSOFF}"                     <7>
320 PRINT"{DOWN RIGHT}TASTE."              <131>
330 BETA$:IFA$=CHR$(13)THEN340:ELSE330     <107>
340 PRINT"{CLEAR}":POKE56,47:V=65280:VOL
B                                          <160>
350 POKEV+18,PEEK(V+18)AND251              <157>
360 POKEV+19,PEEK(V+19)AND3OR48            <154>
370 FORT=832TO849:READA:POKET,A:PR=PR+A:
NEXTT                                      <107>
380 SYS832                                 <236>
390 FORT=12800TO13080:READA:IFA>-1THENPO
KET,A:NEXT                                 <246>
400 FORX=1TO16:READA:POKEX+293,A:NEXT      <221>
410 DATA162,0,189,0,208,157,0,48,189,0,2
09,157,0,49,202,208,241,96                 <190>
420 DATA000,000,024,012,007,003,001,000    <129>
430 DATA170,000,000,000,170,255,170,000    <194>
440 DATA170,001,015,015,170,255,170,127    <182>
450 DATA170,192,240,252,170,255,170,255    <105>
460 DATA170,000,000,000,128,224,224,128    <35>
470 DATA170,000,000,000,001,007,007,001    <5>
480 DATA170,001,015,063,170,255,170,255    <68>
490 DATA170,192,240,240,170,255,170,254    <66>
500 DATA170,000,000,000,170,255,170,000    <8>
510 DATA000,000,024,048,224,192,128,000    <87>
520 DATA000,000,000,000,192,255,192,000    <194>
530 DATA000,000,000,000,003,255,003,000    <50>
540 DATA000,000,000,000,242,255,242,000    <28>
550 DATA170,000,012,063,255,255,127,031    <242>
560 DATA170,000,048,252,255,255,254,248    <180>
570 DATA000,024,066,214,153,214,066,024    <46>
580 DATA255,066,036,024,024,036,066,255    <17>
590 DATA024,024,126,255,255,255,126,000    <32>
600 DATA153,214,066,214,153,036,170,066    <233>
610 DATA000,000,000,000,066,153,170,036    <138>
```

```
620 DATA000,000,000,000,036,170,153,066   <188>
630 DATA128,192,048,220,129,214,170,153   <177>
640 DATA153,170,214,129,220,048,192,128   <95>
650 DATA001,002,005,036,214,153,129,214   <95>
660 DATA214,129,153,214,036,005,002,001   <209>
670 DATA153,066,214,036,128,000,000,000   <66>
680 DATA129,024,002,059,002,036,128,002   <219>
690 DATA000,000,000,255,128,192,224,240   <97>
700 DATA000,000,000,001,003,007,015,031   <77>
710 DATA000,000,024,036,036,024,000,000   <81>
720 DATA255,170,170,255,170,170,255,170   <65>
730 DATA002,000,000,012,000,001,128,002   <18>
740 DATA-1                                <86>
750 DATA46,0,51,46,1,51,46,2,51,46,3,51   <249>
760 DATA110,5,51,96                       <148>
770 POKE0,15                              <77>
780 PRINT"{DOWN2}LOAD{DOWN2 HOME}";:POKE
1319,13:POKE1320,13:POKE239,2             <41>
```

Teil 2:

```
100 REM ********************************  <211>
110 REM *     -- AIRWOLF-SPIEL --       *  <93>
120 REM ********************************  <231>
130 REM *         WRITTEN BY            *  <168>
140 REM **                             **  <103>
150 REM ***    ANDREAS POLEFKA        ***  <103>
160 REM ****                         ****  <103>
170 REM ***    5470 ANDERNACH 1       ***  <241>
180 REM **                             **  <143>
190 REM *          [C] 1986             *  <142>
200 REM ********************************  <55>
210 COLOR0,1                              <97>
220 COLOR4,3,3                            <137>
230 COLOR1,2                              <123>
240 REM ***************                   <165>
250 REM ** TITELBILD **                   <97>
260 REM ***************                   <186>
270 POKE65298,PEEK(65298)AND251           <72>
280 COLOR1,8,3:PRINT"{CLEAR DOWN3 RIGHT2
 SPACE}πππ{SPACE2}π{SPACE}πππ{SPACE2}π{S
PACE2}π{SPACE2}π{SPACE2}πππ{SPACE2}π{SPA
CE4}πππππ":H=1                           <162>
290 COLOR1,8,4:PRINT"{RIGHT2}π{SPACE3}π{
SPACE}π{SPACE}π{SPACE2}π{SPACE}π{SPACE2}
π{SPACE2}π{SPACE}π{SPACE3}π{SPACE}π{SPAC
E4}π"                                    <73>
300 COLOR1,8,5:PRINT"{RIGHT2}π{SPACE3}π{
SPACE}π{SPACE}π{SPACE2}π{SPACE}π{SPACE2}
π{SPACE2}π{SPACE}π{SPACE3}π{SPACE}π{SPAC
E4}π"                                    <89>
310 COLOR1,8,6:PRINT"{RIGHT2}π{SPACE3}π{
SPACE}π{SPACE}π{SPACE2}π{SPACE}π{SPACE2}
π{SPACE2}π{SPACE}π{SPACE3}π{SPACE}π{SPAC
E4}π"                                    <105>
320 COLOR1,8,7:PRINT"{RIGHT2}πππππ{SPACE
}π{SPACE}πππ{SPACE2}π{SPACE2}π{SPACE2}π{
SPACE}π{SPACE3}π{SPACE}π{SPACE4}ππ"      <112>
330 COLOR1,8,6:PRINT"{RIGHT2}π{SPACE3}π{
SPACE}π{SPACE}πππ{SPACE3}π{SPACE2}π{SPACE
2}π{SPACE}π{SPACE3}π{SPACE}π{SPACE4}π"   <191>
340 COLOR1,8,5:PRINT"{RIGHT2}π{SPACE3}π{
SPACE}π{SPACE}π{SPACE}π{SPACE2}π{SPACE2}
π{SPACE2}π{SPACE}π{SPACE3}π{SPACE}π{SPAC
E4}π"                                    <162>
350 COLOR1,8,4:PRINT"{RIGHT2}π{SPACE3}π{
SPACE}π{SPACE}π{SPACE2}π{SPACE}π{SPACE2}
π{SPACE2}π{SPACE}π{SPACE3}π{SPACE}π{SPAC
E4}π"                                    <133>
360 COLOR1,8,3:PRINT"{RIGHT2}π{SPACE3}π{
SPACE}π{SPACE}π{SPACE2}π{SPACE2}πππ{SPACE
}πππ{SPACE3}πππ{SPACE2}ππππ{SPACE}π":COLO
R1,2                                     <196>
370 PRINT"{DOWN RIGHT6 RVSON}WRITTEN{SPA
CE}BY{SPACE}ANDREAS{SPACE}POLEFKA{RVSOFF
}"                                       <96>
380 PRINT"{RIGHT15 RVSON}[C]{SPACE}1986{
RVSOFF}"                                 <203>
390 PRINT"{RIGHT11 DOWN RVSON}PUNKTE";PU
;"TIME";INT(ZI);"{RVSOFF}":CLR           <44>
400 PRINT"{RIGHT11 DOWN4 RVSON FLASHON}P
RESS{SPACE}FIRE{SPACE}BUTTON{FLASHOFF RV
SOFF}"                                   <88>
410 A$="¯_| |↖{SPACE3}¯_| |↖{SPACE3}¯_| |↖{
SPACE3}¯_| |↖{SPACE3}¯_| |↖{SPACE3}¯_| |↖{S
PACE3}"                                  <110>
420 RESTORE530:H=1                       <44>
430 READX,Y:Y=Y*10:IFJOY(1)>127THEN580   <121>
440 PRINT"{HOME DOWN18}";MID$(A$,H,40);  <194>
450 IFH=8THENH=1                         <45>
460 H=H+1                                <238>
470 IFX=-1THEN420                        <95>
480 SOUND1,X,Y:SOUND2,X-2,Y              <200>
490 GOTO430                              <53>
500 REM ***************                  <116>
510 REM ** TITELMUSIK **                 <237>
520 REM ***************                  <137>
530 DATA739,1,685,1,596,1,724,1,643,2,68
5,1,596,1,739,1,724,4,739,1,685,1        <192>
540 DATA596,1,724,1,643,2,413,1,453,1,51
6,1,516,1,571,1,596,4                    <12>
550 DATA834,1,881,1,874,1,834,4,770,1,81
0,1,834,4,834,1,881,1,874,1,834,4        <146>
560 DATA770,1,810,1,834,4,834,1,881,1,87
4,1,834,4                                <149>
570 DATA770,1,834,1,770,1,834,1,770,1,83
4,1,770,1,834,1,770,1,-1,0               <57>
580 PRINT"{CLEAR}":PU=0:ZI=0             <49>
590 CHAR1,0,1,"{SPACE}PUNKTE{SPACE20}TIM
E{SPACE9}":LE=1:COLOR1,10,7              <17>
600 PRINT"♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠
♠♠♠♠♠♠♠♠♠♠"                               <113>
610 PRINT"_,{SPACE19}♠_____X♠_____o_
"                                        <24>
620 PRINT"X♠{SPACE10}|{SPACE9}♠_♥_____
____♠Xo_♠"                               <209>
630 PRINT"__TTTTTT{SPACE15}|♠_____X♠| ___
_ |__"                                   <67>
640 PRINT"__TTTT{SPACE18}|♠_____o♠
X"                                       <90>
650 PRINT"__T||{SPACE21}____♠X♥_____"    <59>
660 PRINT"__T{SPACE23}| | |♠_____o,_ |"  <148>
670 PRINT"X♠X{SPACE13}o__,{SPACE9}♠_____
♠♠_ |X"                                  <59>
680 PRINT"_♠ |,{SPACE10}|o_♠X_,{SPACE12}♠
_____ |"                                 <220>
690 PRINT"__X♥{SPACE10}| _____|{SPACE11}
|♠X♠___"                                 <85>
700 PRINT"___X{SPACE9}o_ |,_____o♥|{SPACE1
0}o_____"                                <68>
710 PRINT"____,|{SPACE6}o____,♠____X♠o,{
SPACE6}o__o____"                         <233>
720 PRINT"_X__♠,{SPACE5}oX_____,| X,|___
,{SPACE5}♠__♠o___"                       <62>
730 PRINT"___,♠,_o |_____X♠_____{S
PACE7}o_♠___"                            <118>
740 PRINT"_____♥_____♠_____,,_X{S
PACE6}|_|o___"                           <90>
750 PRINT"_____♠X_____||IX♠_____| |{S
PACE7}o♥____"                            <115>
760 PRINT"_____X♠_X{SPACE5}♠_____
{SPACE8}_ |____"                         <102>
770 PRINT"__X | |♠__X |_ |X{SPACE9}♠X__X{SPA
CE6}|♠ |X____"                           <97>
780 PRINT"__♥{SPACE8}|{SPACE11}| |{SPACE8
```

```
}o♠♠X♥__"                                                    <46>
790 PRINT"X_ |{SPACE7} || oX♠| {SPACE7} |{SPA
CE7} |o__o_____"                                             <254>
800 PRINT"_| {SPACE5} |o_o| __ |__X♠_| | o_ | ,|
{SPACE4}_X♠o_X♠__"                                           <42>
810 PRINT"X|{SPACE6}♠_____X♠_____,
| o_ ,| o_____"                                               <188>
820 PRINT"__{SPACE7}_____
_____{HOME}":COLOR1,2:GOTO1080                               <184>
830 CHAR1,0,1,"{SPACE}PUNKTE{SPACE20}TIM
E":LE=2:COLOR1,10,7                                          <13>
840 PRINT"{HOME DOWN2}♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠
♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠♠"                                        <18>
850 PRINT"_♥{SPACE7}_ ,|__♠X_____|
|_| |♠ |X_____"                                              <138>
860 PRINT"_♠{SPACE8} | |♠_____X |{S
PACE5}♠__o_ |_"                                              <103>
870 PRINT"_,{SPACE12} |_X |♠_ |♠ |X♠ |X{SPACE
6}| | |♠_♠_"                                                <134>
880 PRINT"__X♠,|{SPACE9}|{SPACE7}|{SPACE
10}|{SPACE2}|__"                                             <248>
890 PRINT"_____,| |♥| ♥o__,| ♥,♥| ♠,X♠| | ♥o,|·
♥{SPACE6} |__"                                               <118>
900 PRINT"___♠X |_____X♠_____ |___X♠| X{
SPACE9}__"                                                   <162>
910 PRINT"_____X |♠__| |X_ | |♠_X |{SPACE2
}|{SPACE8}|o__♠_"                                            <168>
920 PRINT"X__ |♠_| ♥X | | | |♠ |_X{SPACE13}o_ ||
_o_♠X_"                                                      <235>
930 PRINT"_|_X{SPACE}|{SPACE5}|{SPACE5}|{
SPACE12} |_X♠__X_____"                                       <179>
940 PRINT"__X{SPACE5}| o____,| ♥,__X♠,o| ♥X♠
____| X♠_____"                                               <121>
950 PRINT"__,{SPACE4}| |♠_____X♠_____X |♠
_♥X | |__X♠X♠____"                                           <162>
960 PRINT"__ |{SPACE7} |♠ |X♠__| X{SPACE10}|
{SPACE7}♠__"                                                 <3>
970 PRINT"___,{SPACE9}|{SPACE10}|♠|_♥ |_o,
{SPACE6}♠_"                                                  <22>
980 PRINT"_____♠♥o,| ♥♥| ♠__Xo,{SPACE8}♠X |
♠X |{SPACE5}♠_"                                               <191>
990 PRINT"_____X_____X{SPACE9}|{
SPACE8}|_"                                                   <31>
1000 PRINT"__X |♠_♥| X |♠ |X | |♠_____| X♠_♥ |_,o
_,{SPACE5}♠__X"                                              <207>
1010 PRINT"__{SPACE4}|{SPACE10}♠___ |♥♠X |
X♠ |__|{SPACE5}o__"                                          <239>
1020 PRINT"_X{SPACE5}o_♥o,{SPACE4}|{SPAC
E4}|{SPACE7}♠| {SPACE7}♠_"                                   <0>
1030 PRINT"_X{SPACE4}| | |♥♠__X | |_X |♠_♥X{S
PACE7}♠X|{SPACE5}♠_"                                         <204>
1040 PRINT"_{SPACE8}|{SPACE6}|{SPACE9}|{
SPACE10}o__"                                                 <215>
1050 PRINT"_ |{SPACE5}|♠| _ |,o_| ♥,♠X o,o,{S
PACE6}o_,oo,_____"                                           <16>
1060 PRINT"__{SPACE7}_____,| ♥o
__,o_____{HOME}":COLOR1,2                                    <219>
1070 POKE4037,81:IFK=1THENPOKE4037-1024,
255                                                          <246>
1080 IFA=3ANDLE=1THENE=3994:B=2:A=23:CHA
R1,B,A,"-♠|—":GOTO1110                                       <184>
1090 IFLE=2ANDA=24THENE=3235:B=3:A=4:CHA
R1,B,A,"-♠|—":GOTO1110                                       <45>
1100 E=3235:B=3:A=4:CHAR1,B,A,"-♠|—"                         <145>
1110 J=JOY(1)                                                <0>
1120 IFLE=1ANDA=24THEN830                                    <82>
1130 IFLE=2ANDA=3THEN590                                     <71>
1140 IFZI<50ANDZI>2ANDE=3235ANDLE=1THEN2
140                                                          <245>
1150 IFZI>50THEN2100                                         <83>
1160 IFK=1THENZI=ZI+.2                                       <121>
1170 GOSUB2080                                               <231>
1180 ONJ+1GOTO1190,1200,1220,1240,1260,1
280,1300,1320,1340                                           <121>
1190 E=E:FORG=0TO10:NEXTG:GOTO1360                           <24>
1200 E=E-40:CHAR1,B,A,"{SPACE5}":A=A-1:B
=B:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN1510                      <91>
1210 CHAR1,B,A,"-♠|—":GOTO1360                               <85>
1220 E=E-39:CHAR1,B,A,"{SPACE5}":A=A-1:B
=B+1:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN151
0                                                            <55>
1230 CHAR1,B,A,"-♠|—":GOTO1360                               <105>
1240 E=E+1:CHAR1,B,A,"{SPACE5}":A=A:B=B+
1:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN1510                       <208>
1250 CHAR1,B,A,"-♠|—":GOTO1360                               <125>
1260 E=E+41:CHAR1,B,A,"{SPACE5}":A=A+1:B
=B+1:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN151
0                                                            <22>
1270 CHAR1,B,A,"-♠|—":GOTO1360                               <145>
1280 E=E+40:CHAR1,B,A,"{SPACE5}":A=A+1:B
=B:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN1510                      <142>
1290 CHAR1,B,A,"-♠|—":GOTO1360                               <166>
1300 E=E+39:CHAR1,B,A,"{SPACE5}":A=A+1:B
=B-1:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN151
0                                                            <138>
1310 CHAR1,B,A,"¯_| |,":GOTO1360                             <205>
1320 E=E-1:CHAR1,B,A,"{SPACE5}":A=A:B=B-
1:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN1510                       <66>
1330 CHAR1,B,A,"¯_| |,":GOTO1360                             <225>
1340 E=E-41:CHAR1,B,A,"{SPACE5}":A=A-1:B
=B-1:IFPEEK(E)<>32ORPEEK(E+4)<>32THEN151
0                                                            <165>
1350 CHAR1,B,A,"¯_| |,":GOTO1360                             <245>
1360 SOUND3,900,3                                            <217>
1370 IFZ>0THEN1400                                           <118>
1380 KU=INT(RND(1)*880)+3192:Z=1                             <173>
1390 IFPEEK(KU)<>32THENZ=0                                   <0>
1400 IFX=1THEN1590                                           <230>
1410 IFX=2THEN1810                                           <201>
1420 IFX=3THEN1880                                           <15>
1430 IFJOY(1)>128THEN1760                                    <101>
1440 IFJOY(1)=128THEN1580                                    <119>
1450 IFZ=1THEN1960                                           <32>
1460 IFZ=2THEN1980                                           <62>
1470 GOTO1110                                                <245>
1480 REM ******************************                     <146>
1490 REM *** VERLUST EINES AIRWOLFS ***                      <71>
1500 REM ******************************                     <166>
1510 FORG=0TO4:POKEE+G,90:NEXTG                              <122>
1520 FORG=1TO3:POKEE-40+G,95:POKEE+40+G,
95:NEXTG                                                     <74>
1530 FORG=1000TO700STEP-5:SOUND3,G,2                         <24>
1540 NEXTG:GOTO280                                           <134>
1550 REM *************                                       <179>
1560 REM *** BOMBE ***                                       <77>
1570 REM *************                                       <199>
1580 D=E+42:X=1:IFPEEK(D)<>32THEN1650                        <12>
1590 POKED,32:D=D+40:IFPEEK(D)<>32THEN16
50                                                           <170>
1600 POKED,76:IFZ=1THEN1960                                  <26>
1610 IFZ=2THEN1980:ELSE1110                                  <135>
1620 REM ****************                                    <205>
1630 REM *** AUFSCHLAG ***                                   <170>
1640 REM ****************                                    <225>
1650 IFPEEK(D)=93THENPU=PU+30:P=32:GOTO1
690                                                          <54>
1660 IFPEEK(D)=79THENPU=PU+150:P=32:Z=0:
GOTO1690                                                     <213>
1670 IFPEEK(D)=81THENPU=PU+300:K=1:P=81:
POKED-1024,255:GOTO1690                                      <103>
1680 P=PEEK(D)                                               <122>
1690 POKED,90:FORG=800TO700STEP-10                           <188>
1700 SOUND3,G,2:NEXTG                                        <183>
1710 POKED,P:X=0                                             <39>
```

```
1720 GOTO1110                                          <240>
1730 REM **************                                <221>
1740 REM *** RAKETE ***                                <190>
1750 REM **************                                <241>
1760 IFPEEK(E)=64THEN1780                              <202>
1770 IFPEEK(E)=69THEN1850                              <227>
1780 D=E+5:X=2                                         <56>
1790 POKED-1,78:SOUND3,600,5:FORSZ=0TO10
:NEXTSZ:POKED-1,68                                     <255>
1800 IFPEEK(D)<>32THEN1650                             <80>
1810 POKED,32:D=D+2:IFPEEK(D-1)<>32THEND
=D-1:GOTO1650                                          <140>
1820 IFPEEK(D)<>32THEN1650                             <100>
1830 POKED,74:IFZ=1THEN1960                            <247>
1840 IFZ=2THEN1980:ELSE1110                            <110>
1850 D=E-1:S=A:T=B-1:X=3                               <99>
1860 POKED+1,77:SOUND3,600,5:FORSZ=0TO10
:NEXTSZ:POKED+1,69                                     <64>
1870 IFPEEK(D)<>32THEN1650                             <150>
1880 POKED,32:D=D-2:IFPEEK(D+1)<>32THEND
=D+1:GOTO1650                                          <175>
1890 IFPEEK(D)<>32THEN1650                             <170>
1900 IFT<1THENX=0:GOTO1110                             <28>
1910 POKED,75:IFZ=1THEN1960                            <76>
1920 IFZ=2THEN1980:ELSE1110                            <190>
1930 REM *******************                           <106>
1940 REM *** FELSBROCKEN ***                           <224>
1950 REM *******************                           <126>
1960 IFPEEK(KU)>63ANDPEEK(KU)<74THEN1510               <210>
1970 IFPEEK(KU)<>32THEN1110                            <171>
1980 POKEKU,32:KU=KU+40:IFPEEK(KU)<>32TH
ENZ=0:GOTO2000                                         <61>
1990 POKEKU,79:Z=2:GOTO1110                            <162>
2000 IFPEEK(KU)>63ANDPEEK(KU)<74THEN1510               <250>
2010 IFPEEK(KU)=74ORPEEK(KU)=75THENPU=PU
+150:X=0:POKEKU,32                                     <70>
2020 H=PEEK(KU)                                        <17>
2030 POKEKU,90:SOUND3,800,10:FORSX=0TO10
:NEXTSX                                                <175>
2040 POKEKU,H:GOTO1110                                 <153>
2050 REM *********************                         <241>
2060 REM * PUNKTE/LEVEL/TIME *                         <233>
2070 REM *********************                         <5>
2080 CHAR1,7,1,STR$(PU)                                <59>
2090 CHAR1,31,1,STR$(INT(ZI)):RETURN                   <240>
2100 FORL=1000TO0STEP-5                                <206>
2110 V=INT(RND(1)*255):SOUND3,L,2                      <95>
2120 POKE65301,V:NEXTL                                 <26>
2130 RUN                                               <228>
2140 CHAR1,8,10,"{RVSON}HERZLICHEN{SPACE
}GLUECKWUNSCH{RVSOFF}":ZI=0:K=0                        <40>
2150 CHAR1,8,12,"{RVSON}SIE{SPACE}HABEN{
SPACE}ES{SPACE}GESCHAFFT{RVSOFF}":PU=PU+
500                                                    <173>
2160 CHAR1,4,14,"{RVSON}UND{SPACE}NUN{SP
ACE}ZUM{SPACE}NAECHSTEN{SPACE}ABENTEUER{
RVSOFF}"                                               <202>
2170 CHAR1,11,16,"{RVSON FLASHON}PRESS{S
PACE}FIRE{SPACE}BUTTOM{RVSOFF FLASHOFF}"               <92>
2180 IFJOY(1)=128THEN590:ELSE2180                      <58>
```

TIPS & TRICKS für den C-16/116

UPDATE für die Beep-Routine

Die Beep-Routine (C-16/C-116) aus Heft 4/86 habe ich abgetippt und mir ist aufgefallen, daß das Maschinenprogramm in einen sehr ungünstigen Bereich gefallen ist. Laut des Beitextes „merkt der Anwender von Existenz und Ablauf der Routine FAST NICHTS". Das „fast nichts" hat mich dann stutzig gemacht! Da Graphik nicht möglich war und obendrein mein 60K Speicher auf 12K wieder absank. Schade, aber die Abhilfe war leicht. Kurz das ROM-Listing nach freiem Speicher durchwühlt und, was mich auch selbst erstaunte, einen Speicherbereich, der für 12 K und auch für 60K zu gebrauchen ist, gefunden: Den Rambereich für Sprach-Synthesizer $ 065E – $ 06EB. Ich habe dann nur noch die Interrupt-Adresse geändert und schon ging es los.

Vorteile:

- **Die Graphik ist für die Zeichner voll einsetzbar**
- **Die Routine ist nach einem RESET ohne viel Mühe mit einem SYS-Befehl wieder aktiv**
- **Der Speicher ist voll erhalten und ohne Einschränkung für BASIC verfügbar**

Nachteil:

- **Wer diesen Bereich schon belegt hat, muß entweder auf die Beep-Routine verzichten oder sich was Neues einfallen lassen.**

Wer die Beep-Routine schon eingegeben hat, braucht nur in Zeile **10005** und **10006** die folgenden Daten ändern: **10005 Data A2,75; 10006 Data A0,06** und die Start- und End-Adresse in Zeile **10030** in **For I = 1630 to 1674** und in Zeile **10110 If P = 5208 ...** und Zeile **0** entfällt. Alles andere bleibt wie gehabt. Das Abspeichern der Routine wird dann über **TEDMON** gemacht.

```
10 FORI=0TO46
20 READA$:A=DEC(A$):POKE1630+I,A:B=B+A
30 NEXT:IFB=5208THENSYS1630
100 REM DATA
110 DATA 78        :REM  SEI
120 DATA A9,7F     :REM  #$7F
130 DATA 8D,10,FF  :REM  STA $FF10
140 DATA A9,B0     :REM  LDA #$B0
150 DATA 8D,0F,FF  :REM  STA $FF0F
160 DATA A2,75     :REM  LDX #$75
170 DATA A0,06     :REM  LDY #$06
180 DATA 8E,14,03  :REM  STX $0314
190 DATA 8C,15,03  :REM  STY $0315
200 DATA 58        :REM  CLI
210 DATA 60        :REM  RTS
220 DATA A5,C6     :REM  LDA $C6
230 DATA C9,40     :REM  CMP #$40
240 DATA F0,08     :REM  BEQ $0683
250 DATA A9,22     :REM  LDA #$22
260 DATA 8D,11,FF  :REM  STA $FF11
270 DATA 4C,0E,CE  :REM  JMP $CE0E
280 DATA A9,00     :REM  LDA #$00
290 DATA 8D,11,FF  :REM  STA $FF11
300 DATA 4C,0E,CE  :REM  JMP $CE0E
310 DATA 00        :REM  BRK
320 DATA 00        :REM  BRK
```

Speichererweiterung für den C-16/116

64 K

Sensationell:
– für weniger als 50 DM
– voll kompatibel zum Urzustand durch Umschalter

Die Erweiterung des Speichers auf 64 K-Byte wird am Beispiel des C-116 besprochen. Da aber die Nummern und Anschlüsse der Chips beim C-16 gleich sind und nur die Lage der Bauelemente anders ist, kann auch dieses Gerät nach dieser Anleitung erweitert werden.
Obwohl der Umbau des Rechners an sich einfach ist, sollte sich dennoch jeder Umbauwillige fragen, ob er auch genügend Erfahrung im Löten feinster Leiterbahnen hat. Abgelöste, verbrannte oder falsch durchtrennte Leiterbahnen versprechen Ärger. Eine doppelseitige Platine zu reparieren, gleicht einem Geduldspiel.
Für Unsichere gibt es einen einfachen und billigen Weg ihre Fähigkeiten zu testen. Löten Sie z. B. einen 7400 Baustein für ca. 2,-- DM auf ein Stück Experimentierplatine. Damit haben Sie geübt und Sie wissen sich und Ihre Fähigkeiten einzuschätzen.
So sind vier Voraussetzungen zum erfolgreichen Basteln unerläßlich:
1. Ein qualifizierter Umrüster
2. Gutes Werkzeug
3. Ausreichend Zeit
4. Noch mehr Geduld

Folgende Werkzeuge sind zu beschaffen:
- Ein kleiner Elektonik-Lötkolben mit feiner Spitze. Leistung ca. 16 Watt
- Elektronik-Lot mit 60% Zinn-Anteil und einem Durchmesser kleiner als 1,5 mm
- Ein kleiner Seitenschneider, möglichst mit planen Schneiden
- Ein feiner Uhrmacher-Schraubenzieher oder ein spitzes scharfes Messer zum Durchtrennen zweier Leiterbahnen
- Ein kleiner Kreuzschlitz-Schraubenzieher
- Ein Bohrer für den Schalterdurchgang
- Eine Entlötpumpe; zur Not tuts auch Entlötlitze

Benötigte Bauteile:
1. Die neuen Speicherelemente: 4 x 64 K dynamische RAMs z. B. von Texas Instruments TMS 4464-15. Dieses Bauteil wird 2 x benötigt.
2. Ein 2fach UM-Schalter. Vom Einbau her sollte es ein Micro-Schalter mit wenig Platzbedarf sein.
3. Feine Litze für die Verbindungen.

Die genannten Bauteiletypen sind keineswegs ein Muß. Es wurden lediglich mit diesen Typen Erfahrungen gesammelt.
Eine mögliche Bezugsquelle findet sich am Schluß. Der Preis für die beiden Speicherbausteine und den Microschalter liegt unter 50,-- DM.

Vorbereitungen:
Bevor Sie anfangen, sich mit dem Innenleben Ihres Rechners zu beschäftigen, sollten Sie stets an einen geerdeten Gegenstand gefaßt haben, um elektrostatische Aufladungen abzuführen. Blitze vom Synthetikpulli sind das Ende des Computers. Mit dem Kreuzschlitz-Schraubenzieher werden nun die vier Gehäuseschrauben an der Vorder- bzw. Rückseite des Rechners entfernt. Beachten Sie, daß die längeren Schrauben später wieder zum Verschrauben der Rückseite verwendet werden. Heben Sie jetzt vorsichtig das Gehäuse-Oberteil ab. Sie müssen dabei einen kleinen Klemmwiderstand überwinden. Sie sehen jetzt ein mit Leiterbahnen versehenes Kunststoffband von der Tastatur im Deckel zu einer Buchse in der Platine reichen. Ziehen Sie das Band (vorsichtig!!) aus der Buchse indem Sie beide Hände auf das Abschirmblech auflegen, mit Daumen und Zeigefinger das Band greifen und nach oben herausziehen. Es geht anfangs schwer, aber bitte trotzdem nicht zu heftig reißen.
Nun werden die Halteschrauben für das Abschirmblech herausgedreht. Achten Sie beim Abnehmen des Bleches darauf, daß die heruntergebogene Lasche nicht verbogen wird. Sie dient der Kühlung des TED-Chips. Bei schlechter Auflage sollte die Lasche evtl. mit etwas Wärmeleitpaste bestrichen werden. Nun schrauben Sie noch die Schrauben zur Platinenhalterung heraus. Jetzt haben Sie Ihren Einplatinen-Computer in der Hand.

Der Umbau:
Sehen Sie nun nach vorne rechts. Dort sind zwei längere IC's (18-polig) mit den Nummern U5 und U6. Dies sind die beiden Speicherbausteine, die ausgewechselt werden müssen. Beim Herausnehmen sollte nicht an eine Erhaltung der Baustei-

Bild 1

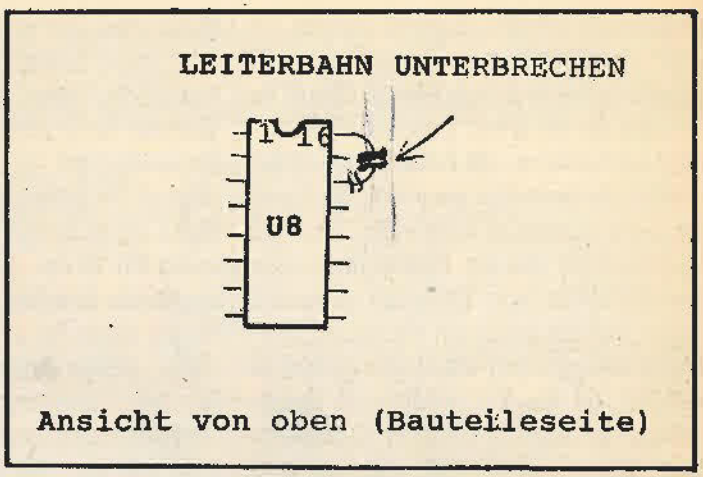


Bild 2

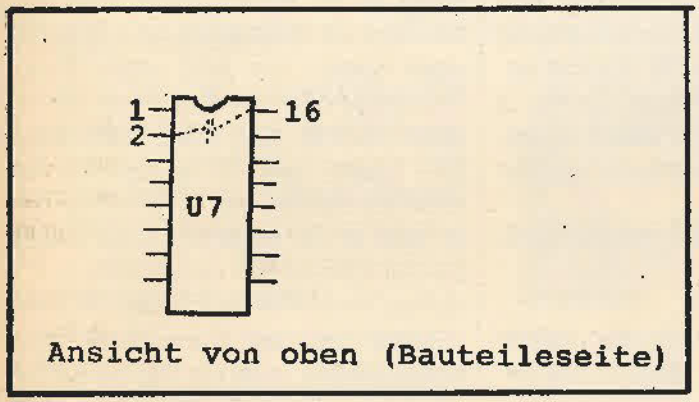


Bild 3a

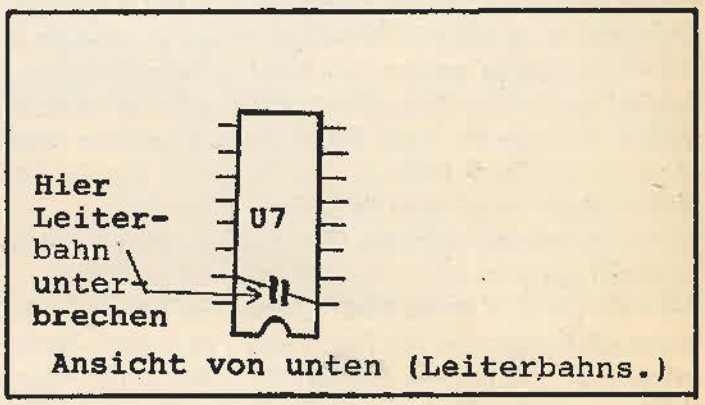


Bild 3b

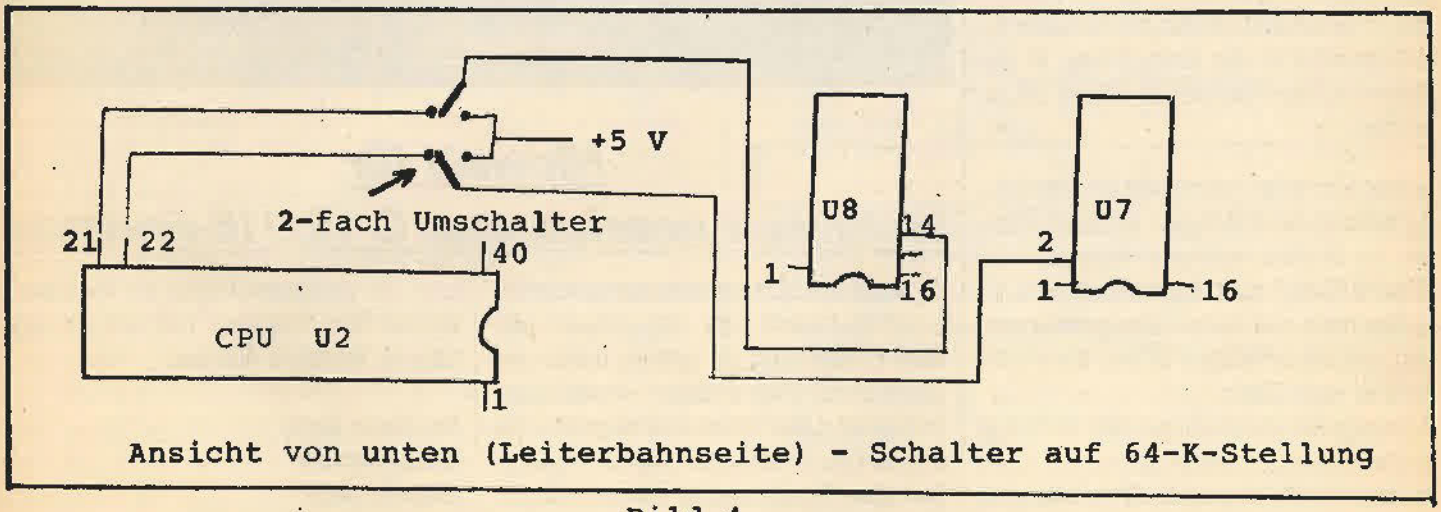


Bild 4

ne gedacht werden. Erstens werden sie nicht mehr benötigt, und zweitens ist die Erhaltung der Platine höchstes Gebot. So knipsen Sie bei den IC's wie auf Bild 1 gezeigt am Gehäuse alle Beinchen ab. Dann werden die Beinchen vorsichtig nach und nach herausgelötet und jeder Lötpunkt anschließend mit der Entlötpumpe oder der Entlötlitze von Lotresten gereinigt. Beim Einlöten brauchen die neuen Bausteine übrigens nur von unten gelötet zu werden, da das Lot nach oben durchdringt. Die Lage der Kerbe im IC muß unbedingt dem auf der Platine aufgedruckten Umriß entsprechen, damit der Speicher richtig gepolt wird.

Nun müssen zwei Verbindungen unterbrochen werden. An U8 ist auf der Bauteile-Seite der Platine die Verbindung zwischen dem Pol 16 und dem Pol 14, wie in Bild 2 dargestellt, zu unterbrechen. An U7 muß auf der Platinen-Unterseite die Verbindung zwischen Pol 2 und Pol 16 getrennt werden. Siehe Bilder 3a und 3b. Achtung: Beim Umdrehen ist die Zählrichtung der Anschlüsse gespiegelt (Bild 3b). Verbinden sie jetzt noch entsprechend Bild 4 die CPU (U2) an den Polen 22 und 21 mit dem Schalter und den Polen 2 und 14 von U7 und U8. Den nun noch benötigten Pluspol können Sie beispielsweise an einer Leiterbahn anschließen, die mit Pin 16 von U7 oder U8 verbunden ist.

Es bleibt im Wesentlichen dem Geschmack eines jeden einzelnen überlassen, wo er den Schalter einbaut. Um jedoch übermäßige Kabellängen zu vermeiden, ist es am einfachsten den Schalter direkt gegenüber der Einbuchtung in der Platine bei den Bausteinen U7/U8 einzupassen.

Sollten Sie daran interessiert sein, ein Eingabe/Ausgabe-Interface anzuschließen, wie es in einer späteren Ausgabe von COMPUTE MIT noch besprochen wird, so sollten noch zwei kurze Kabel gelötet werden, um ein unnötiges Öffnen des Rechners zu vermeiden.

Achtung: Der Baustein, an dem die Kabel angelötet werden sollen, hat im C-116 die Nummer U101 und im C-16 die Nummer U16!!!

Verbunden werden muß der Pin 15 von U101 (nicht benutzt) mit Pin Z des Expension-Ports. Wenn der Rechner mit der Bauteile-Seite nach unten und mit dem Expensionport zu Ihnen auf dem Tisch liegt ist Pin Z der vierte Pin von rechts der Reihe vorne zum Platinenrand. Weiterhin sollte Pin 16 des gleichen Bausteines für eventuelle Erweiterungen auf Pin AA, das ist der dritte Pin von rechts der gleichen Reihe, gelegt werden.

Vor dem Zusammenbau sollten Sie Netz- und Antennen- bzw. Videokabel anschließen und den Rechner einschalten. Der Text sollte jetzt... 60671 BYTES FREE lauten. Wenn nicht: Keine Panik! Legen Sie den neuen Schalter um und drücken Sie die Reset-Taste. Erscheint immer noch nicht das Gewünschte oder etwa ein Wert von ca. 28 kbyte, so muß die Verkabelung überprüft werden. Wenn Ihr Rechner einwandfrei funktioniert, bauen Sie ihn in umgekehrter Reihenfolge wieder zusammen. Vorsicht beim Folienleiterband der Tastatur.

Damit ist der Umbau an der Platine des C-116 beendet.

Ein Hinweis für Insider:

Im Graphic-Modus liegt in der 12kbyte Version das Programm **VOR** dem Graphikbereich, in der 60kbyte Version wird beim Graphik-Aufruf alles **HINTER** den Graphikbereich transferiert, auch Maschinenprogramme. In dieser Tatsache liegt der Grund für die Notwendigkeit eines Umschalters auf Originalzustand, z. B. bei einigen Spielen. Um noch einmal darauf hinzuweisen: Es gibt Programme, die mit keiner noch so guten POKE-Kombination zum Laufen gebracht werden können, wenn der Rechner auf 60kbyte-free umgerüstet ist und KEINEN Schalter besitzt, um auf 12kbyte-free umzuschalten.

Sollten Sie den Wunsch haben, an Ihrem „Gummi"-Computer eine andere Tastatur anzuschließen, so ist das Parallel zur Originaltastatur möglich. Wenn dafür Interesse besteht, so schreiben Sie uns.

Michael Lausch

In der nächsten Ausgabe:

VERSITALE INTERFACE

(Eingabe – Ausgabe)

im Selbstbau

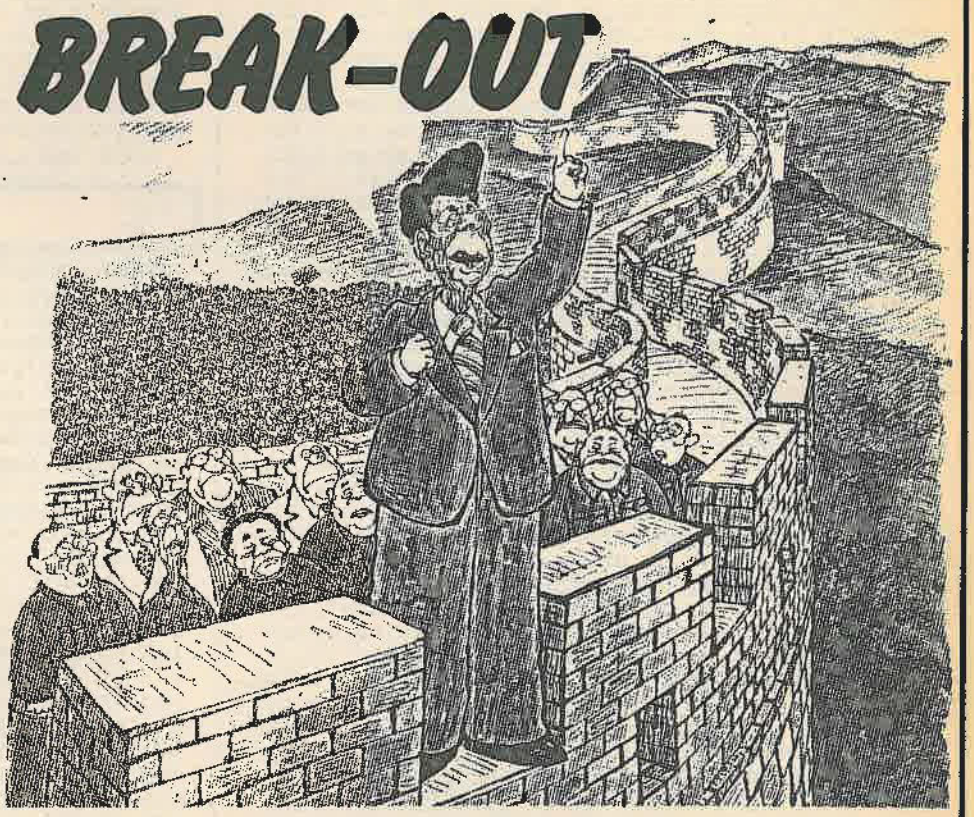
Bezugsquelle für die Bauteile: Fa. Elektronik Richter, Rheinstraße 85, 6200 Wiesbaden, Tel. 0 61 21 / 37 71 17.

Hinweis für technisch unerfahrene C-16/116-Freunde

Um auch den in Hardware unerfahreneren C-16/116-Freund die Möglichkeit der RAM-Erweiterung zu geben, haben wir uns noch mit einer weiteren Firma zusammengetan. Diese Firma übernimmt für Sie den Umbau Ihres Systems, wenn Sie Ihren Computer einsenden, zu einem Preis von 79,— DM inclusive Bauteile und Arbeitslohn. Die Umbauzeit liegt in der Regel bei 1 Woche. Wer Interesse hat, wendet sich bitte an folgende Adresse:

Wolf-Dieter Herm
Bischofsgasse 9
6940 Weinheim
Tel.: 0 62 01 / 6 91 89

Speed ist Trumpf auf dem C-16/116



Bei diesem Spiel ist Geschicklichkeit gefragt. Schlagen Sie mit Hilfe eines Schlägers den Ball gegen eine Mauer aus Steinen. Trifft der Ball einen dieser Steine, werden Ihnen 5 Punkte gutgeschrieben. Bei einem Bonus-Stein sind es sogar 100 Punkte. Trifft der Ball aber das grüne Herz, so springt das Spiel in das nächste Level, in dem die Spielgeschwindigkeit erhöht wird. Insgesamt müssen Sie sich durch 5 Level schlagen. Das Spiel ist vollkommen in Maschinensprache geschrieben. Gespielt wird mit **„Z" = links** und **„/" = rechts** oder mit **Joystick**.

Eingabehinweise:

Zunächst Teil 1 abtippen und speichern, dann Teil 2 abtippen und speichern usw. Die Teile möglichst kurz, d. h. ohne Leerzeichen zwischen den Befehlen abtippen. Zum Starten Teil 1 einladen und starten, Teil 2 und Teil 3 werden automatisch nachgeladen und gestartet.

```
1 REM ---------------------------------  <250>
2 REM -- B R E A K - O U T ( C 1 6 ) --  <62>
3 REM ---------------------------------  <252>
4 REM --    GESCHRIEBEN VON :        --  <151>
5 REM --    CARSTEN REINKE           --  <167>
6 REM --    1000 BERLIN 21           --  <89>
7 REM ---------------------------------  <0>
8 :                                      <66>
10 GRAPHIC 0,1                           <144>
20 FORA=1TO18:READB$:B=DEC(B$):FE=FE+B:P
OKE831+A,B:NEXTA                         <122>
30 DATA A2,00,BD,00,D0,9D,00,30,BD,00,D1
,9D,00,31,CA,D0,F1,60                    <38>
40 IFFE<>2115THENPRINT"DATA-FEHLER{SPACE
}30":STOP:ELSE SYS832                    <90>
50 FOR A = 1 TO 17                       <234>
60 READ PO                               <80>
70 PS=12288+8*PO                         <213>
80 FOR AN = 0 TO 7                       <78>
90 READ BY                               <112>
100 POKE PS+AN,BY:PR=PR+BY               <194>
110 NEXT AN,A                            <81>
120 DATA 0,215,215,85,85,85,85,215,215   <173>
140 DATA 64,171,171,171,255,207,255,243,
255                                      <112>
150 DATA 65,171,171,171,255,234,234,234,
255                                      <44>
160 DATA 66,255,119,87,87,84,87,223,207  <142>
170 DATA 67,255,195,195,235,235,215,215,
255                                      <176>
180 DATA 68,213,90,90,213,213,90,90,213  <77>
190 DATA 69,85,170,170,85,85,150,150,85  <170>
200 DATA 70,87,165,165,87,87,165,165,87  <199>
210 DATA 71,255,252,255,255,85,85,215,21
5                                        <158>
220 DATA 72,0,0,0,7,15,31,31,31          <18>
230 DATA 73,0,0,0,255,255,255,255,255    <239>
240 DATA 74,0,0,0,224,240,248,248,248    <255>
250 DATA 75,31,31,31,31,31,31,31,31      <114>
260 DATA 76,248,248,248,248,248,248,248,
248                                      <72>
270 DATA 79,248,248,248,240,224,0,0,0    <197>
280 DATA 78,255,255,255,255,255,0,0,0    <81>
290 DATA 77,31,31,31,15,7,0,0,0          <212>
300 IF PR<>19847 THEN PRINT"DATA-FEHLER{
SPACE}120-290":STOP                      <126>
310 SCNCLR                               <31>
320 PRINT"{DOWN2}LOAD"                   <4>
330 PRINT"{HOME}";                       <16>
340 POKE 1319,13:POKE1320,82:POKE1321,21
3:POKE 1322,13:POKE 239,4:END            <208>
  ENDE DES LISTINGS
```

Teil 2:

```
10 POKE52,31:POKE56,31:CLR:SCNCLR        <218>
20 FORA=8192TO9200STEP23:FE=0:FORB=0TO22
:READC$:C=DEC(C$):FE=FE+CAND255          <201>
30 POKEA+B,C:NEXTB:READF$:F=DEC(F$):IFF<
>FETHEN40:ELSENEXTA:GOTO50               <132>
40 PRINT"DATA-FEHLER";60+INT((A-8192)/24
):STOP                                   <0>
50 PRINT"{CLEAR DOWN2}LOAD{HOME}";:POKE1
319,13:POKE1320,82:POKE1321,213:POKE1322
,13:POKE239,4:END                        <121>
60 DATA A9,0B,85,D1,A9,07,85,D3,A9,D8,85
,D0,85,D2,AE,3D,03,E0,00,D0,01,60,A5,E3  <147>
70 DATA D0,18,69,28,90,04,E6,D1,E6,D3,85
,D0,85,D2,CA,4C,11,20,CE,3C,03,60,EE,CB  <150>
```

```
80 DATA 3C,03,60,CE,3D,03,60,EE,3D,03,60
,AD,3E,03,C9,01,D0,12,20,29,20,AD,3C,87   <178>
90 DATA 03,C9,01,D0,05,A9,02,8D,3E,03,4C
,65,20,C9,02,D0,F9,20,2D,20,AD,3C,03,D9   <20>
100 DATA C9,1C,D0,EF,A9,01,8D,3E,03,AD,3
F,03,C9,01,D0,10,20,31,20,AD,3D,03,C9,DC  <185>
110 DATA 01,D0,05,A9,02,8D,3F,03,60,C9,0
2,D0,FB,20,35,20,AD,3D,03,C9,17,D0,F1,49  <79>
120 DATA A9,01,8D,3F,03,60,AD,40,03,C9,0
0,F0,F8,20,A0,20,CE,40,03,4C,90,20,A2,09  <48>
130 DATA 06,FE,E8,0C,BD,E8,0C,C9,3A,D0,0
8,A9,30,9D,E8,0C,CA,10,EE,60,20,00,20,56  <27>
140 DATA AC,3C,03,B1,D0,C9,3F,90,19,C9,4
4,B0,15,C9,40,D0,12,A9,05,8D,40,03,20,78  <151>
150 DATA 3D,23,A9,20,91,D0,A9,02,8D,3F,0
3,60,C9,41,D0,12,A9,05,8D,40,03,20,3D,2B  <46>
160 DATA 23,A9,40,91,D0,A9,02,8D,3F,03,6
0,C9,42,D0,0D,20,D6,23,A9,20,91,D0,A9,1B  <220>
170 DATA 02,8D,3F,03,60,A9,64,8D,40,03,2
0,3D,23,A9,20,91,D0,A9,02,8D,3F,03,60,92  <208>
180 DATA 20,00,20,AD,3C,03,18,69,28,A8,B
1,D0,C9,3F,90,EF,C9,45,B0,EB,AD,3E,03,1C  <177>
190 DATA C9,01,D0,08,A9,02,8D,3E,03,4C,4
0,21,C9,02,D0,05,A9,01,8D,3E,03,AD,3C,C9  <198>
200 DATA 03,C9,01,D0,08,A9,02,8D,3E,03,4
C,58,21,C9,1C,D0,05,A9,01,8D,3E,03,AD,C2  <106>
210 DATA 3F,03,C9,01,D0,08,A9,02,8D,3F,0
3,4C,70,21,C9,02,D0,05,A9,01,8D,3F,03,54  <138>
220 DATA AD,3D,03,C9,01,D0,08,A9,02,8D,3
F,03,4C,88,21,C9,17,D0,05,A9,01,8D,3F,29  <71>
230 DATA 03,20,39,20,4C,39,20,AD,3D,03,C
9,17,D0,05,A9,12,8D,42,03,60,20,00,20,F0  <53>
240 DATA AD,3C,03,18,69,50,A8,B1,D0,C9,4
4,D0,09,A9,01,8D,3E,03,8D,3F,03,60,C9,3C  <79>
250 DATA 45,D0,0B,A9,00,8D,3E,03,A9,01,8
D,3F,03,60,C9,46,D0,0A,A9,02,8D,3E,03,D2  <213>
260 DATA A9,01,8D,3F,03,60,A9,20,AE,43,0
3,9D,99,0F,9D,9A,0F,9D,9B,0F,60,AE,43,B9  <144>
270 DATA 03,A9,44,9D,99,0F,A9,45,9D,9A,0
F,A9,46,9D,9B,0F,A9,79,9D,99,0B,9D,9A,39  <68>
280 DATA 0B,9D,9B,0B,60,A5,C6,C9,0C,D0,0
C,AE,43,03,E0,01,90,04,CA,8E,43,03,60,31  <168>
290 DATA C9,37,D0,FB,AE,43,03,E0,19,B0,0
4,E8,8E,43,03,60,AC,43,03,20,FF,21,98,52  <6>
300 DATA CD,43,03,F0,F3,A8,AD,43,03,8C,4
3,03,A8,20,D2,21,98,8D,43,03,4C,E1,21,D7  <103>
310 DATA 20,00,20,AD,3C,03,18,69,50,A8,B
1,D0,C9,45,D0,04,EE,44,03,60,C9,44,F0,9A  <169>
320 DATA 04,C9,46,D0,F7,A9,00,8D,44,03,6
0,AD,44,03,C9,02,90,F8,AE,3C,03,A9,47,DB  <72>
330 DATA 9D,90,0D,A9,00,8D,44,03,60,AC,3
C,03,B1,D0,C9,47,D0,F6,C0,0A,90,0B,A9,67  <117>
340 DATA 01,8D,3E,03,A9,20,99,90,0D,60,A
9,02,8D,3E,03,A9,20,99,90,0D,60,A0,18,BE  <79>
350 DATA A2,00,CA,D0,FD,88,D0,F8,60,AD,F
B,BF,AA,78,8E,08,FF,AD,08,FF,8E,08,FF,50  <124>
360 DATA CD,08,FF,D0,F2,58,49,FF,A8,29,0
F,AA,BD,F0,BF,C0,0F,90,02,09,80,A8,C0,7E  <125>
370 DATA 07,D0,03,4C,05,22,C9,03,D0,03,4
C,15,22,60,A2,04,DE,B0,0D,BD,B0,0D,C9,53  <5>
380 DATA 2F,D0,0D,A9,39,9D,B0,0D,CA,10,E
E,A9,12,8D,42,03,60,20,00,20,AD,3C,03,29  <199>
390 DATA 18,69,28,A8,A9,20,91,D0,60,20,0
0,20,AD,3C,03,18,69,28,A8,A9,00,91,D0,62  <133>
400 DATA A9,79,91,D2,60,AD,43,03,84,D4,2
0,A4,22,A5,D4,CD,43,03,F0,0F,AC,43,03,93  <62>
410 DATA 8D,43,03,20,D2,21,8C,43,03,20,E
1,21,60,AD,3D,03,C9,02,F0,04,60,4C,E1,73  <34>
420 DATA 21,AD,40,03,8D,0E,FF,AD,12,FF,0
9,01,8D,12,FF,AD,11,FF,09,18,8D,11,FF,8C  <220>
430 DATA 4C,90,20,A5,D5,CD,43,03,D0,01,6
0,A9,03,8D,10,FF,A9,00,8D,0F,FF,AD,11,04  <1>
440 DATA FF,09,28,8D,11,FF,AD,43,03,85,D
5,60,A9,00,8D,0E,FF,8D,0F,FF,8D,10,FF,F4  <201>
450 DATA 8D,11,FF,AD,12,FF,29,FC,8D,12,F
F,60,20,F1,22,20,39,20,20,76,22,20,21,23  <127>
460 DATA 22,20,13,23,20,3F,22,20,9B,21,2
0,61,22,20,8E,21,20,14,21,20,32,23,20,31  <254>
470 DATA B5,20,20,E1,21,20,D7,22,EA,EA,E
A,20,56,23,20,00,23,20,99,22,20,76,23,3E  <16>
480 DATA 20,99,22,AD,42,03,C9,12,D0,BD,2
0,E1,21,4C,76,23,A9,12,8D,42,03,A9,6F,E1  <81>
490 DATA 8D,50,03,60,FF,FF,FF,FF,FF,FF,F
F,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,FF,2D  <40>
  ENDE DES LISTINGS
```

Teil 3:

```
9 FORA=13056TO13063:POKEA,0:NEXTA                <102>
10 TRAP1040                                      <147>
20 DEFFNPU(X)=PEEK(3310-X)-48                    <83>
30 FORA=1TO6:H(A)=16-A:H$(A)="C.REINKE";
NEXTA:HI=H(1)                                    <130>
40 PUDEF"0":LE=3:SC=0:GOTO760                    <37>
50 VOL8:FORA=1TO3                                <226>
60 VOL8:FORA=1TO3                                <236>
70 SOUND1,685,8:SOUND2,854,8                     <46>
80 SOUND1,739,12:SOUND2,881,12                   <113>
90 SOUND1,770,4:SOUND2,897,4                     <36>
100 SOUND1,798,6:SOUND2,911,6                    <221>
110 NEXTA:SOUND1,810,40:SOUND2,917,40            <41>
120 GRAPHIC 0,1                                  <254>
130 POKE 65302,92:POKE 65303,66                  <72>
140 POKE 65298 , PEEK (65298) AND 251            <197>
150 POKE 65299 , PEEK (65299) AND 3 OR 4
8                                                <51>
160 POKE8858,24                                  <65>
170 RESTORE180:FORA=0TO19:READB$:POKE832
+A,DEC(B$):NEXTA                                 <199>
180 DATA A9,60,A2,00,9D,00,0C,9D,00,0D,9
D,00,0E,9D,00,0F,CA,D0,F1,60                     <144>
190 COLOR 1,3,4:COLOR 0,1:COLOR 4,1:SYS5
5432:SYS 832                                     <63>
200 PRINT"└────────────────────────────┘
"                                                <89>
210 FOR A = 1 TO 23                              <116>
220 PRINT"{WHITE SPACE28}";:COLOR1,3,4:
PRINT"└"                                         <45>
230 NEXT A                                       <234>
240 PRINT"┌────────────────────────────┐
";                                               <152>
250 IFJA=1THEN310:ELSECOLOR1,10,7:CHAR1,
1,1,"{SPACE}♠{SPACE}♠{SPACE}♠{SPACE}♠{SP
ACE}─{SPACE}♠{SPACE}♠{SPACE}♠{SPACE}♠{SP
ACE}♠{SPACE}─{SPACE}♠{SPACE}♠{SPACE}♠"           <14>
260 CHAR1,1,2,"♠{SPACE}♠{SPACE}─{SPACE}♠
{SPACE}♠{SPACE}♠{SPACE}♠{SPACE}♠{SPACE}─
{SPACE}♠{SPACE}♠{SPACE}♠{SPACE}♠{SPACE}♠
{SPACE}"                                         <133>
270 CHAR1,1,3,"{SPACE}♠{SPACE}♠{SPACE}─{
SPACE}♠{SPACE}♠{SPACE}♠{SPACE}│{SPACE}♠{
SPACE}♠{SPACE}♠{SPACE}─{SPACE}♠{SPACE}♠{
SPACE}♠"                                         <133>
280 CHAR1,1,4,"♠{SPACE}♠{SPACE}♠{SPACE}♠
{SPACE}─{SPACE}♠{SPACE}─{SPACE}─{SPACE}♠
{SPACE}♠{SPACE}♠{SPACE}─{SPACE}─{SPACE}♠
{SPACE}"                                         <83>
290 CHAR1,1,5,"{SPACE}♠{SPACE}♠{SPACE}♠{
SPACE}♠{SPACE}♠{SPACE}♠{SPACE}─{SPACE}♠{
SPACE}♠{SPACE}♠{SPACE}♠{SPACE}♠{SPACE}♠{
SPACE}─"                                         <171>
300 REM ---- BLANK = SHIFT + SPACE ----          <101>
310 COLOR 1,2,7:CHAR1,32,4,"SCORE{SHIFTS
PACE}:":CHAR1,32,5,"":PRINTUSING"#######
";SC                                             <140>
```

```
320 CHAR1,32,9,"ZEIT{SHIFTSPACE}:":CHAR1
,32,10,"00999"                          <160>
330 CHAR 1,32,14,"LEBEN{SHIFTSPACE}:":CH
AR1,32,15,"{LEFT SHIFTSPACE}"           <147>
340 B$=STR$(LE):L=LEN(B$)-1:P$=RIGHT$(B$
,L):PRINTP$                             <240>
350 CHAR1,30,20,"BREAK-OUT":CHAR1,30,21,
"VON":CHAR1,30,22,"CARSTEN":CHAR1,32,23,
"REINKE"                                <18>
360 IFJA=1THENRETURN                    <212>
370 REM ***** SPIEL *****               <18>
380 POKE 828,10:POKE 829,10:POKE 830,1:P
OKE 831,1:POKE 834,0:POKE 835,10:POKE836
,0                                      <31>
390 POKE 65287,PEEK(65287)OR16          <8>
400 RESTORE410:FORA=0TO7:READB:POKE12544
+A,B:NEXTA                              <148>
410 DATA 255,252,255,255,207,255,243,255 <232>
420 SYS 9101                            <253>
430 POKE 2021,23:POKE 2022,1:POKE2023,1:
POKE2024,28                             <63>
440 POKE 65287,PEEK(65287) AND 239      <64>
450 FOR A=0TO7:POKE 12544+A,0:NEXT A    <73>
460 SCNCLR                              <181>
470 SC=0:ZA=1:FORA=0TO6:SC=SC+FNPU(A)*ZA
:ZA=ZA*10:NEXTA                         <225>
480 IFPEEK(848)<>111THENLE=LE-1:IFLE>0TH
EN170                                   <19>
490 IFPEEK(848)=111THENPOKE 8858,PEEK(88
58)-4                                   <214>
500 IFPEEK(8858)>9ANDLE>0THEN170        <40>
510 COLOR1,8:CHAR1,5,2,"DEIN{SPACE}SCORE
{SPACE}:{SPACE}":PRINTUSING"#######";SC <13>
520 IFSC<=HITHENCHAR1,3,4,"{WHITE RVSON
SPACE}B{SPACE}R{SPACE}E{SPACE}A{SPACE}K{
SPACE}-{SPACE}O{SPACE}U{SPACE}T{SPACE}C-
16{SPACE RVSOFF YELLOW}"                <97>
530 IFSC>HITHENCHAR1,3,4,"{WHITE RVSON F
LASHON SPACE}N{SPACE}E{SPACE}U{SPACE}E{S
PACE}R{SPACE2}R{SPACE}E{SPACE}K{SPACE}O{
SPACE}R{SPACE}D{SPACE FLASHOFF RVSOFF YE
LLOW}":HI=SC                            <106>
540 VOL8:SOUND1,739,10:SOUND2,881,10    <21>
550 SOUND1,770,10:SOUND2,897,10         <94>
560 SOUND1,798,10:SOUND2,911,10         <216>
570 SOUND1,810,10:SOUND2,917,10         <238>
580 SOUND1,770,30:SOUND2,897,30         <176>
590 FORA=1TO6:IFSC>H(A)THENGOTO600:ELSEN
EXTA:GOTO660                            <45>
600 FORB=5TOASTEP-1:H(B+1)=H(B):H$(B+1)=
H$(B):NEXTB:H$(A)="":H(A)=SC            <160>
610 CHAR1,5,6,"DEIN{SPACE}NAME{SPACE}:{S
PACE}?"                                 <128>
620 FORC=1TO8:POKE239,0                 <212>
630 GETKEYK$:K=ASC(K$):IFK>31ANDK<96THEN
650                                     <186>
640 GOTO630                             <208>
650 H$(A)=H$(A)+K$:CHAR1,17,6,H$(A):NEXT
C:A=6:NEXTA                             <212>
660 CHAR1,15,6,"":PRINT"{DOWN}":COLOR1,6
,7:FORA=1TO6:PRINTTAB(3);H$(A),         <120>
670 PRINTUSING"######";H(A):PRINT:NEXTA <32>
680 COLOR1,5,7:CHAR1,3,21,"BITTE{SPACE}D
RUECKE{SPACE}EINE{SPACE}TASTE":COLOR1,7,
7                                       <174>
690 POKE 239,0:ZA=1:P$="{SPACE3}T{SPACE}
O{SPACE}P{SPACE}-{SPACE}S{SPACE}I{SPACE}
X{SPACE2}O{SPACE}F{SPACE2}***{SPACE}B{SP
ACE}R{SPACE}E{SPACE}A{SPACE}K{SPACE}-{SP
ACE}O{SPACE}U{SPACE}T{SPACE}***{SPACE7}" <215>
700 CHAR1,15,18,"":POKE 2021,19:POKE 202
2,8:POKE2023,15:POKE2024,15             <148>
710 PRINTMID$(P$,ZA,1);:ZA=ZA+1:IFZA>LEN
(P$)THENZA=1                            <179>
720 GETA$:IFA$<>""THEN750               <114>
730 FORA=1TO100:NEXTA                   <204>
740 GOTO 710                            <48>
750 LE=3:SC=0:RETURN                    <47>
760 JA=1:GOSUB 120:JA=0                 <207>
770 POKE 2021,23:POKE 2022,1:POKE2023,1:
POKE2024,28                             <149>
780 COLOR1,4,7:PRINT"{CLEAR}";TAB(8);"{D
OWN}WILLKOMMEN{SPACE}ZU"                <43>
790 PRINT"{DOWN SPACE2}***{SPACE}B{SPACE
}R{SPACE}E{SPACE}A{SPACE}K{SPACE}-{SPACE
}O{SPACE}U{SPACE}T{SPACE}***"           <245>
800 PRINT"{DOWN SPACE}ANLEITUNG{SPACE}={
SPACE}'SPACE',"                         <239>
810 PRINT"{DOWN SPACE}SONST{SPACE}SPIEL.
"                                       <134>
820 GETKEYA$:IFA$<>CHR$(32)THEN1010     <56>
830 PRINT"{DOWN SPACE}DAS{SPACE}SPIEL{SP
ACE}GEHT{SPACE}UEBER"                   <173>
840 PRINT"{DOWN SPACE}5{SPACE}LEVEL.{SPA
CE}SIE{SPACE}HABEN{SPACE}3"             <71>
850 PRINT"{DOWN SPACE}SCHLAEGER{SPACE}ZU
R{SPACE}VERFUEGUNG."                    <68>
860 PRINT"{DOWN SPACE}VERSUCHEN{SPACE}SI
E{SPACE}ZU{SPACE}VERHIN-"               <163>
870 PRINT"{DOWN SPACE}DERN,{SPACE}DASS{S
PACE}DER{SPACE}BALL{SPACE}DEN"          <104>
880 PRINT"{DOWN SPACE}BODEN{SPACE}BERUEH
RT.":GOSUB 1020                         <139>
890 PRINT"{DOWN SPACE}TRIFFT{SPACE}DER{S
PACE}BALL{SPACE}DIE{SPACE}MAUER"        <217>
900 PRINT"{DOWN SPACE}KOENNEN{SPACE}SIE{
SPACE}5{SPACE}PUNKTE"                   <204>
910 PRINT"{DOWN SPACE}IHR{SPACE}EIGEN{SP
ACE}NENNEN,{SPACE}DURCH"                <21>
920 PRINT"{DOWN SPACE}EINEN{SPACE}BONUS-
PUNKT{SPACE}KASSIEREN"                  <152>
930 PRINT"{DOWN SPACE}SIE{SPACE}WEITERE{
SPACE}100{SPACE}PUNKTE.":GOSUB 1020     <131>
940 PRINT"{DOWN SPACE}UM{SPACE}IN{SPACE}
DAS{SPACE}NAECHSTE,"                    <130>
950 PRINT"{DOWN SPACE}SCHNELLERE{SPACE}L
EVEL{SPACE}ZU"                          <54>
960 PRINT"{DOWN SPACE}GELANGEN,{SPACE}MU
SS{SPACE}DER{SPACE}BALL"                <253>
970 PRINT"{DOWN SPACE}DAS{SPACE}GRUENE{S
PACE}HERZ{SPACE}TREFFEN.":GOSUB1020     <191>
980 PRINT"{DOWN SPACE}DRUECKEN{SPACE}SIE
{SPACE}'RETURN'{SPACE}UM"               <79>
990 PRINT"{DOWN SPACE}ZU{SPACE}SPIELEN{S
PACE}!!!"                               <152>
1000 GETKEYA$:IFA$<>CHR$(13)THEN1000    <205>
1010 GOSUB50:GOTO760                    <85>
1020 PRINT"{DOWN SPACE}DRUECKEN{SPACE}SI
E{SPACE}EINE{SPACE}TASTE"               <31>
1030 GETKEY A$:PRINT"{UP SPACE27 UP2}":R
ETURN                                   <201>
1040 POKE 65287,PEEK(65287)AND239       <155>
1050 POKE 65298,196:POKE65299,208       <113>
1060 COLOR0,2:COLOR1,1:COLOR4,15,6:PRINT <34>
1070 SYS55432:PRINTERR$(ER);"{SPACE}ERRO
R{SPACE}IN"EL8                          <181>
1080 HELP                               <38>
```

Aufgrund der Bauanleitung einer Speichererweiterung ist der Programmteil beim C-16/116 diesmal nicht so umfangreich wie gewohnt. Wir bitten um Verständnis!

6502-Assembler-Kurs

Teil 14

STA – STORE ACCUMULATOR IN MEMORY (Speichere Akkumulator in Speicher)

Dies ist das Gegenstück zum LDA-Befehl. Er speichert den Inhalt des Akkumulators an einer bestimmten Speicherstelle ab. Wir haben diesen Befehl schon mehrmals in unseren Beispielen verwendet; da er sicherlich neben LDA der wichtigste Befehl des 6502 ist. Er bietet die gleichen Speicher-Adressierungsmöglichkeiten wie der ADC-Befehl mit der Ausnahme, daß eine unmittelbare Adressier-Möglichkeit nicht vorhanden ist.

Hex	Adressierung	Bytes	Syntax
81	Indirekt, vor-indiziert mit X	2	STA (addr),x
85	Null-Seite (direkt)	2	STA addr
89	Nicht verwendet		
8D	Absolut (direkt)	3	STA addr
91	Indirekt, nach indiziert mit Y	2	STA (addr),y
95	Null-Seite indiziert mit X	2	STA addr,x
99	Absolut indiziert mit Y	3	STA addr,y
9D	Absolut indiziert mit X	3	STA addr,x

76543210
N VBDIZC
nicht beinflußt

STX – Store Index Register X in Memory (Speichere X-Register in Speicher)

Auch diesen Befehl haben wir schon häufiger in unseren Beispielen angewandt. Er stellt das Gegenstück zu LDX da, und arbeitet somit genauso wie STA, mit dem Unterschied, daß hier das X-Register im Speicher abgelegt wird.
Beachten Sie, daß es hier keine indizierten Adressier-Arten unter Verwendung des Indexregisters X gibt. Es gibt auch keine absolute indizierte Adressierung. STX und LDX sind die einzigen Befehle, die die indizierte Nullseiten-Adressierung mit Indexregister Y verwendet. Auch hier werden keine Status-Bits beeinflußt.

Hex	Adressierung	Bytes	Syntax
86	Null-Seite (direkt)	2	STX addr
8E	Absolut (direkt)	3	STX addr
96	Null Seite indiziert mit Y	2	STX addr,Y
9E	Nicht verwendet		

76543210
N VBDIZC
nicht beinflußt

STY – Store Index Register Y in Memory (Speichere Indexregister Y in den Speicher)

Dieser Befehl dient ebenfalls zum Ablegen eines Registers. Er benutzt die gleichen Adressierungsmöglichkeiten wie der STX-Befehl, legt jedoch den Inhalt des Y-Registers im Speicher ab. Die Status-Flags werden auch hier nicht beeinflußt.

Hex	Adressierung	Bytes	Syntax
84	Null-Seite (direkt)	2	STY addr
8C	Absolut (direkt)	3	STY addr
94	Null Seite indiziert mit X	2	STY addr,x
9C	Nicht verwendet		

76543210
N VBDIZC
nicht beinflußt

TAX Transfer Accumulator to Index-Register X (Übertrage Inhalt von Akku nach X-Register)

Neben den bekannten Registern „LADEBEFEHLEN" gibt es beim 6502, sowie beim 6510 noch die sogenannten Transfer-Anweisungen. Diese veranlassen den Prozessor, den Inhalt eines Registers in ein anderes zu übertragen. Das Ursprungsregister wird dadurch nicht beeinflußt. Der TAX-Befehl überträgt den Inhalt des Akkumulators in das X-Register. Transfer-Befehle haben den Vorteil, daß diese keinen Operanten benötigen und dadurch nur 1 Byte belegen. Auch die Ausführungsgeschwindigkeit der Befehle macht sich der geschulte Programmierer oft zu Nutze. In der Praxis werden Transfer-Befehle häufig verwandt um ein Register zwischenzuspeichern, vorallem wird hier der Akkumulator gerettet, da er als einziger Arithmetische Operationen durchführen kann.

Hex	Adressierung	Bytes	Syntax
AA	Impliziert	1	TAX

7	6	5	4	3	2	1	0
N		V	B	D	I	Z	C
x							x

TAY Transfer Accumulator to Index-Register Y (Übertrage Akkumulator in Y-Register)

Dieser Befehl arbeitet auf die gleich Weise wie der TAX-Befehl, jedoch wird hier als Ziel das Y-Register verwendet.

Hex	Adressierung	Bytes	Syntax
A8	Impliziert	1	TAY

7	6	5	4	3	2	1	0
N		V	B	D	I	Z	C
x							x

TSX Transfer Stackpointer to Index Register X (Übertrage Stackzeiger in X-Register)

Auch für den Stackpointer gibt es einen Transfer Befehl. Dieser überträgt den Zeiger, welcher auf den Stack zeigt, in das X-Register. Dadurch läßt sich durch die indizierte Adressierung direkt auf den Stack zugreifen. Die Programmiertechnik sollte jedoch nur von etwas erfahreneren Programmierern genutzt werden, da falsche Stackmanipulationen schnell zum Systemabsturz führen können.

Hex	Adressierung	Bytes	Syntax
BA	Impliziert	1	TSX

7	6	5	4	3	2	1	0
N		V	B	D	I	Z	C
x							x

TXA Transfer Index-Register X to Accumulator (Übertrage X-Register in Akkumulator)

Natürlich besitzt der Prozessor auch Transfer-Befehle in umgekehrter Richtung. Der TXA-Befehl überträgt den Inhalt des X-Register's in den Akkumulator.

Hex	Adressierung	Bytes	Syntax
8A	Impliziert	1	TXA

7	6	5	4	3	2	1	0
N		V	B	D	I	Z	C
x							x

TXS Transfer Index-Register X to Stackpointer (Übertrage X-Register in Stackzeiger)

Genauso ist die Umkehrung für TSX möglich. Der TXS-Befehl lädt den Stackpointer mit dem Inhalt des X-Registers. Dadurch lassen sich beispielsweise Rücksprungadressen manipulieren oder löschen. Auch dieser Befehl sollte mit Vorsicht verwendet werden.

Hex	Adressierung	Bytes	Syntax
9A	Impliziert	1	TXS

7	6	5	4	3	2	1	0
N		V	B	D	I	Z	C
nicht beeinflußt							

Literatur:

- 6502 Microcomputer-Programmierung, Peter Heuer, Hofacker Verlag
- 6502 Programmierung in Assembler, Lance A. Leventhal
- 64 Intern, Angershausen, Becker, Englisch, Gerits, Data-Becker Buch
- 6502/65c02, Christian Persson, Heinz Heise Verlag
- Butterfield's Lehrbuch, J. Butterfield, Hanser-Verlag

„Poker“ mit dem Schneider CPC 464/664 und 6128

In fast jedem Western oder Kriminalfilm hat dieses Spiel seinen festen Platz und erfreut sich großer Beliebtheit, spannungsgeladene Szenen zu untermauern. Pokern gehört einfach dazu, wenn es darum geht, Nervenkitzel zu erzeugen.

Dieses Nervenkitzeln können Sie nun zusammen mit anderen Mitspielern oder aber allein mit Ihrem Computer erleben. Poker macht aus Ihrer „Hackerstube“ eine „Spielhölle“.

Gespielt wird Poker ausschließlich mit dem Joystick oder den Cursortasten. Joystick nach oben und unten oder Cursor nach oben und unten wählt aus, ob man mit 52 oder mit 32 Karten spielt. Bei 52 Karten werden generell die Gewinne verdoppelt.

Das bedeuten die Copy- und die Cursortasten:

Bet:	▲▼	Einsatz im Bereich von 1-50 verändern
	●	Karten werden ausgegeben
Hold:	►◄	zum Steuern des Balkens
	▲▼	Karte wird gesetzt bzw. wieder freigegeben
	●	freie Karten werden ersetzt
Next turn:	●	führt zurück zu 'Bet'
GEWINN:	●	Gewinn wird 1-mal ausgegeben
	►◄	führt zu Risiko 1 bzw. Risiko 2
Risk I:	►◄	wählt die Farbe; bei entsprechender Farbe der Zufallskarte wird der Gewinn 2mal ausgegeben, wenn nicht, geht er verloren
Risk II:	◄►	steuert den Balken unter die gewünschte Farbe
	●	zieht eine Karte; bei richtiger Wahl wird der Gewinn 4mal ausgegeben, ansonsten geht er verloren

Eine Besonderheit stellt die gute Grafik dieses Programmes dar. Dem Autor ist hier eine gute Version des Pokerspielens geglückt.

```
10 REM ******************************
15 REM **     SCHNEIDER CPC 464     **
20 REM **---------------------------**
25 REM **   P  O  K  E  R   (plus)  **
30 REM **---------------------------**
35 REM **   W r i t t e n    b y :  **
40 REM **                           **
45 REM **     Serge Hommel          **
50 REM **     19,rte.de Wilwerdange **
55 REM ** 9952 Drinklange           **
60 REM **                           **
65 REM **        G.D. LUXEMBOURG    **
70 REM **---------------------------**
75 REM ** Tel.: (00352)/98701       **
80 REM ******************************
100 '
110 REM ****************************
120 REM ** Titelbild - Titelmelodie **
130 REM ****************************
140 BORDER 0:INK 0,0:INK 4,2:INK 5,26:IN
K 6,6:INK 7,7:INK 8,13:INK 9,9:MODE 0
150 PEN 7:PRINT" HOMULUS - SOFTWARE":LOC
ATE 7,3:PRINT"presents":LOCATE 5,23:PEN
8:PRINT"published by",,"C O M P U T E  M
 I T";
160 PEN 6:LOCATE 9,19:PRINT STRING$(4,14
3):PEN 5:LOCATE 9,20:PRINT STRING$(4,143
):PEN 4:LOCATE 9,21:PRINT STRING$(4,143)
170 ENT 1,1,-120,1,1,120,1
180 RESTORE 190:FOR t=1 TO 44:READ ton:S
OUND 1,ton,15,15,,1:NEXT
190 DATA 169,179,169,159,169,179,169,213
,169,179,169,142,169,179,169,213,169,179
,169,142,169,179,169,126,169,179,169,142
,169,179,169,213,169,179,169,213,169,179
,169,142,169,179,169,213
200 FOR i=1 TO 500:NEXT
210 FOR i=1 TO 31:INK 1,i
220 PLOT 140,130:DRAW 140,330:DRAW 500,3
30:DRAW 500,270:DRAW 160,270:DRAW 160,13
0:DRAW 140,130:MOVE 160,290:DRAW 160,310
:DRAW 480,310:DRAW 480,290:DRAW 160,290:
MOVE 200,130:DRAW 200,210:DRAW 260,210:D
RAW 260,130:DRAW 200,130:MOVE 220,150
230 DRAW 220,190:DRAW 240,190:DRAW 240,1
50:DRAW 220,150:MOVE 280,130:DRAW 280,25
0:DRAW 300,250:DRAW 300,170:DRAW 340,210
```

```
:DRAW 340,190:DRAW 320,170:DRAW 340,150:
DRAW 340,130:DRAW 310,160:DRAW 300,150:D
RAW 300,130:DRAW 280,130:MOVE 360,130
240 DRAW 360,210:DRAW 420,210:DRAW 420,1
70:DRAW 380,170:DRAW 380,150:DRAW 420,15
0:DRAW 420,130:DRAW 360,130:MOVE 380,180
:DRAW 380,200:DRAW 400,200:DRAW 400,180:
DRAW 380,180:MOVE 440,130:DRAW 440,210:D
RAW 460,210:DRAW 460,190:DRAW 500,210
250 DRAW 500,190:DRAW 460,170:DRAW 460,1
30:DRAW 440,130:PEN 1:LOCATE 13,10:PRINT
"PLUS":LOCATE 17,5:PRINT CHR$(64)
260 FOR y=3 TO 14 STEP 2:SOUND 1,(y*i),3
,15:NEXT:NEXT:FOR i=1 TO 1500:NEXT
270 REM **********************
280 REM ** Symboldefinition **
290 REM **********************
300 SYMBOL AFTER 159:RESTORE 310:FOR i=1
60 TO 238:READ a,b,c,d,e,f,g,h:SYMBOL i,
a,b,c,d,e,f,g,h:NEXT i
310 DATA 204,204,51,51,204,204,51,51,16,
56,124,254,254,16,56,,,108,254,254,254,1
24,56,16,56,56,16,214,254,214,16,56,16,5
6,124,254,124,56,16,,,,,,,,1,3,,,16,16,5
6,124,255,255,,,,,,,,128,15,31,63,63,63,
31,7,3,224,240,248,248,248,240,192,128
320 DATA 1,,,,1,3,,,255,214,16,56,255,25
5,,,,,,,,128,,,,,3,7,15,31,31,63,,,131,1
99,239,239,255,255,,,,128,224,240,240,24
8,31,31,15,7,1,,,,255,255,255,255,255,25
4,254,124,240,240,224,192,,,,,124,56,56,
16,16,,,,,,24,60,126,255,255,126,6,15
330 DATA 31,63,63,31,15,6,60,24,153,255,
255,153,24,24,96,240,248,252,252,248,240
,96,,,,,,1,,,60,60,60,126,255,255,,,,,16
,16,56,56,124,254,1,3,15,63,15,3,1,,255,
255,255,255,255,255,255,254,,128,224,248
,224,128,,,8,28,62,127,127,8,28,,,54
340 DATA 127,127,127,62,28,8,28,28,8,107
,127,107,8,28,8,28,62,127,62,28,8,,,121,
59,31,15,11,8,15,,195,231,255,255,102,,2
55,,158,220,248,240,208,16,240,15,8,41,7
4,89,40,8,40,255,,153,219,153,24,24,66,2
40,16,148,210,154,20,16,20,74,82,36,1
350 DATA 24,60,126,223,60,129,153,24,36,
,,129,82,74,36,128,24,60,126,251,143,199
,163,161,176,176,168,172,255,255,255,255
,126,24,24,126,241,227,197,133,13,13,21,
53,172,170,234,234,235,251,251,251,90,12
6,126,90,126,,,,117,85,87,215,215,223
360 DATA 223,223,7,15,31,31,31,30,60,59,
255,255,223,143,7,1,,28,128,192,224,224,
240,240,240,112,56,124,116,244,244,244,2
46,242,64,64,64,96,80,,136,113,120,124,1
26,126,126,111,111,199,242,242,227,224,2
36,223,233,18,3,6,24,244,,195,255,255
370 DATA 7,7,7,7,51,251,251,75,36,72,127
,143,135,135,199,255,153,153,165,255,66,
126,68,124,36,18,254,241,225,225,227,255
,255,251,245,251,245,234,245,234,100,60,
36,189,173,185,169,185,255,223,175,223,1
75,87,175,87,,,135,255,73,41,38,16,,124
380 DATA 255,131,69,69,56,,,,194,254,36,
40,200,16,17,15,8,19,20,19,48,32,255,139
,10,10,136,10,12,8,16,224,160,160,128,16
0,192,128,64,224,32,120,112,64,113,31,12
,4,29,21,52,101,194,,208,80,80,80,64,80,
32,,,63,127,145,245,253,255,124,,7,255
390 DATA 253,85,85,223,249,,224,240,120,
126,253,248,240,62,31,143,207,39,227,17,
240,3,7,143,143,223,254,252,248,224,195,
140,159,32,63,64,127
400 FOR i=1 TO 50:PRINT:NEXT
410 REM **********************
420 REM ** Bildschirmaufbau **
430 REM **********************
440 as=65:te$="10":hs$=STRING$(3,143)+CH
R$(128)+STRING$(3,143):h$="H O L D":hd$=
"        "
450 DATA Royal flush,250,Straight flush,
150,Four of a kind,75,Full house,40,Flus
h,20,Straight,8,Three of a kind,3,Two pa
irs,1
460 RESTORE 450:FOR i=1 TO 8:READ typ$(i
),w(i):NEXT
470 MODE 1:INK 1,23:INK 2,6:INK 3,13:PEN
 1:sc=100:bet=1:DIM contr(54):DIM f(15)
480 PEN 1:LOCATE 5,10:PRINT CHR$(240);":
 As-K-D-B-10-9-8-7-6-5-4-3-2":LOCATE 5,1
2:PRINT CHR$(241);": As-K-D-B-10-9-8-7"
490 IF JOY(0)=1 OR INKEY(0)=0 THEN RAM=1
:g=2:GOTO 520
500 IF JOY(0)=2 OR INKEY(2)=0 THEN RAM=6
:g=1:GOTO 520
510 GOTO 490
520 CLS:WINDOW#6,1,39,11,21:PAPER#6,3:CL
S#6:WINDOW#7,2,38,24,24:PEN#7,1:CLS#7
530 PEN 3:PRINT CHR$(150)STRING$(21,154)
CHR$(156)SPACE$(3)CHR$(150)STRING$(11,15
4)CHR$(156),CHR$(149)SPACE$(21)CHR$(149)
;:PEN 2:PRINT CHR$(150)CHR$(154)CHR$(156
);:PEN 3:PRINT CHR$(149)SPACE$(11)CHR$(1
49)
540 PRINT CHR$(149)SPACE$(21)CHR$(149);:
PEN 2:PRINT CHR$(149)"P"CHR$(149);:PEN 3
:PRINT CHR$(151)STRING$(11,154)CHR$(157)
,CHR$(149)SPACE$(21)CHR$(149);:PEN 2:PRI
NT CHR$(149)"O"CHR$(149);:PEN 3:PRINT CH
R$(149)SPACE$(11)CHR$(149)
550 PRINT CHR$(149)SPACE$(21)CHR$(149);:
PEN 2:PRINT CHR$(149)"K"CHR$(149);:PEN 3
:PRINT CHR$(147)STRING$(11,154)CHR$(153)
,CHR$(149)SPACE$(21)CHR$(149);:PEN 2:PRI
NT CHR$(149)"E"CHR$(149)SPACE$(2);:PEN 3
:PRINT CHR$(150);
560 PRINT STRING$(7,154)CHR$(156),CHR$(1
49)SPACE$(21)CHR$(149);:PEN 2:PRINT CHR$
(149)"R"CHR$(149)SPACE$(2);:PEN 3:PRINT
CHR$(149)SPACE$(7)CHR$(149),CHR$(149)SPA
CE$(21)CHR$(149);:PEN 2:PRINT CHR$(147)C
HR$(154)CHR$(153)SPACE$(2);
570 PEN 3:PRINT CHR$(151)STRING$(7,154)C
HR$(157),CHR$(149)SPACE$(21)CHR$(149)SPA
CE$(5)CHR$(149)SPACE$(7)CHR$(149),CHR$(1
47)STRING$(21,154)CHR$(153)SPACE$(5)CHR$
(147)STRING$(7,154)CHR$(153)
```

```
580 LOCATE 1,23:PRINT CHR$(150)STRING$(3
7,154)CHR$(156),CHR$(149)SPACE$(37)CHR$(
149),CHR$(147)STRING$(37,154)CHR$(153);:
FOR i=1 TO 8:LOCATE 2,i+1:PRINT typ$(i):
LOCATE 17,i+1:PRINT":":NEXT
590 PEN 2:LOCATE 28,2:PRINT" S C O R E "
:LOCATE 30,7:PRINT" B E T "
600 PEN 1:LOCATE 30,4:PRINT USING "#####
#";sc:LOCATE 33,9:PRINT USING "##";bet
610 WINDOW#1,1,7,12,20:WINDOW#2,9,15,12,
20:WINDOW#3,17,23,12,20:WINDOW#4,25,31,1
2,20:WINDOW#5,33,39,12,20
620 REM ********************
630 REM ** Steuerprogramm **
640 REM ********************
650 FOR i=1 TO 54:contr(i)=0:NEXT:FOR i=
1 TO 5:hold(i)=0:NEXT:hold=0:FOR i=1 TO
33 STEP 8:LOCATE i,22:PRINT hd$:NEXT:CLS
#7
660 FOR i=1 TO 5:PAPER#i,1:PEN#i,3:SOUND
 2,240,10,13,0,0,1:CLS#i:PRINT#i,STRING$
(63,127);:NEXT
670  GOSUB 910
680 FOR w=1 TO 5
690 IF hold(w)=1 THEN 780
700 RANDOMIZE TIME
710 ra=INT(RND(6)*5)
720 IF ra<1 OR ra>4 THEN GOTO 710
730 ran=INT(RND(6)*14)
740 IF ran<ram OR ran>13 THEN GOTO 730
750 IF contr(ran*ra)=1 THEN GOTO 710
760 GOSUB 2270
770 contr(ran*ra)=1
780 NEXT
790 IF hold=1 THEN 820
800 GOSUB 1090
810 IF hold=1 THEN 680
820 GOSUB 1280
830 GOSUB 1540
840 IF sc=0 THEN FOR i=1 TO 3000:NEXT:GO
TO 2780
850 CLS#7:PEN#7,1:SOUND 1,1200,30,15:LOC
ATE #7,15,1:PRINT#7,"NEXT TURN"
860 IF JOY(0)<>16 AND INKEY(9)=-1 THEN 8
60
870 IF JOY(0)=16 OR INKEY(9)=0 THEN 650
880 REM *************
890 REM ** Einsatz **
900 REM *************
910 PEN#7,1:LOCATE#7,15,1:PRINT#7,"B   E
   T"
920 IF bet>=sc THEN bet=sc:sc=0:FOR i=1
TO 10:SOUND 1,40,10,15:SOUND 1,30,5,15:N
EXT:GOTO 1040
930 sc=sc-bet:GOTO 1040
940 PEN 1:LOCATE 33,9:PRINT USING "##";b
et:LOCATE 30,4:PRINT USING "######";sc
950 PEN 3:FOR i=1 TO 8:LOCATE 18,i+1:PRI
NT USING "#####";k(i):NEXT
960 IF JOY(0)=16 OR INKEY(9)=0 THEN RETU
RN
970 IF bet>=50 OR sc=0 THEN GOTO 990
980 IF JOY(0)=1 OR INKEY(0)=0 THEN bet=b
et+1:sc=sc-1:SOUND 1,(1040-bet*10),10,15
:GOTO 1040
990 IF bet<=1 THEN GOTO 960
1000 IF JOY(0)=2 OR INKEY(2)=0 THEN bet=
bet-1:sc=sc+1:SOUND 1,(1040-bet*10),10,1
5:GOTO 1040
1010 IF JOY(0)<>1 AND JOY(0)<>2 OR INKEY
(0)+INKEY(2)=-2 THEN GOTO 940
1020 IF bet >50 THEN bet=50
1030 IF bet <1 THEN bet=1
1040 FOR i=1 TO 8:k(i)=w(i)*bet*g:NEXT
1050 GOTO 940
1060 REM *******************
1070 REM ** Karten halten **
1080 REM *******************
1090 CLS#7:PEN#7,1:PRINT#7,"
  H  O  L  D":l=1:hold=1
1100 IF JOY(0)=16 OR INKEY(9)=0 THEN 119
0
1110 PEN#6,0:LOCATE#6,l,11:PRINT#6,hs$;
1120 IF l>25 THEN l=33:GOTO 1140
1130 IF JOY(0)=8 OR INKEY(1)=0 THEN FOR
f=1 TO 100:NEXT:SOUND 1,100,3,15:LOCATE#
6,l,11:PRINT#6,hd$;:l=l+8:GOTO 1110
1140 IF l<8 THEN l=1:GOTO 1160
1150 IF JOY(0)=4 OR INKEY(8)=0 THEN FOR
f=1 TO 100:NEXT:SOUND 1,100,3,15:LOCATE#
6,l,11:PRINT#6,hd$;:l=l-8:GOTO 1110
1160 IF JOY(0)=1 OR INKEY(0)=0 THEN PEN
2:SOUND 1,200,10,15:LOCATE 1,22:PRINT h$
:hold((l+7)/8)=1
1170 IF JOY(0)=2 OR INKEY(2)=0 THEN LOCA
TE 1,22:SOUND 1,400,10,15:PRINT hd$:hold
((l+7)/8)=0
1180 GOTO 1100
1190 LOCATE#6,l,11:PRINT#6,hd$;
1200 GOSUB 1220
1210 RETURN
1220 FOR i=1 TO 5:IF hold(i)=0 THEN PEN#
i,3:CLS#i:SOUND 2,240,10,13,0,0,1:PRINT#
i,STRING$(63,127);
1230 NEXT
1240 RETURN
1250 REM ****************
1260 REM ** Auswertung **
1270 REM ****************
1280 aa=aa(1):ab=aa(2):ac=aa(3):ad=aa(4)
:ae=aa(5):ba=bb(1):bb=bb(2):bc=bb(3):bd=
bb(4):be=bb(5):sa=aa+ab+ac+ad+ae:maa=MAX
(aa,ab,ac,ad,ae):mia=MIN(aa,ab,ac,ad,ae)
1290 IF ba=bb AND bb=bc AND bc=bd AND bd
=be THEN GOTO 1400
1300 IF (aa=ab AND ab=ac AND ac=ad) OR (
aa=ab AND ab=ac AND ac=ae) OR (aa=ab AND
 ab=ad AND ad=ae) OR (aa=ac AND ac=ad AN
D ad=ae) OR (ab=ac AND ac=ad AND ad=ae)
THEN typ=3:GOTO 1430
1310 IF (aa=ab AND ac=ad AND ad=ae) OR (
aa=ac AND ab=ad AND ad=ae) OR (aa=ad AND
 ab=ac AND ac=ae) OR (aa=ae AND ab=ac AN
D ac=ad) OR (aa=ac AND ac=ae AND ab=ad)
OR (aa=ac AND ac=ad AND ab=ae) THEN typ=
4:GOTO 1430
```

```
1320 IF (ab=ac AND aa=ad AND ad=ae) OR (
aa=ab AND ab=ae AND ac=ad) OR (aa=ab AND
 ab=ad AND ac=ae) OR (aa=ab AND ab=ac AN
D ad=ae) THEN typ=4:GOTO 1430
1330 IF (aa=ab AND (ab=ac OR ab=ad OR ab
=ae)) OR (aa=ac AND (ac=ad OR ac=ae)) OR
 (ad=ae AND (ad=aa OR ad=ab OR ad=ac)) O
R (ab=ac AND (ac=ad OR ac=ae)) THEN typ=
7:GOTO 1430
1340 IF aa<> ab AND aa<>ac AND aa<>ad AN
D aa<>ae THEN GOTO 1360
1350 IF (aa=ab AND (ac=ad OR ac=ae OR ad
=ae)) OR (aa=ac AND (ab=ad OR ab=ae OR a
d=ae)) OR (aa=ad AND (ab=ac OR ab=ae OR
ac=ae)) OR (aa=ae AND (ab=ac OR ab=ad OR
 ac=ad)) THEN typ=8:GOTO 1430
1360 IF (ab=ac AND ad=ae) OR (ab=ad AND
ac=ae) OR (ab=ae AND ac=ad) THEN typ=8:G
OTO 1430
1370 IF aa=ab OR aa=ac OR aa=ad OR aa=ae
 OR ab=ac OR ab=ad OR ab=ae OR ac=ad OR
ac=ae OR ad=ae OR (maa-mia)<>4 THEN GOTO
 1390
1380 IF sa>10 AND sa<60 AND (sa/5-INT(sa
/5))=0 THEN typ=6
1390 GOTO 1430
1400 IF sa=55 AND maa-mia=4 THEN typ=1:G
OTO 1430
1410 IF sa>10 AND sa<55 AND sa/5-INT(sa/
5)=0 AND maa-mia=4 THEN typ=2:GOTO 1430
1420 typ=5
1430 IF typ=0 THEN 1500
1440 RESTORE 1450:FOR t=1 TO 12:READ ton
:SOUND 1,ton,10,15:NEXT
1450 DATA 268,225,1,225,1,253,225,201,22
5,253,225,225
1460 a$=typ$(typ)
1470 win=k(typ)
1480 PEN 2:LOCATE 2,(typ+1):PRINT a$:LOC
ATE 18,typ+1:PRINT USING "#####";win
1490 GOSUB 1720
1500 RETURN
1510 REM ********************
1520 REM ** Gewinn ausgeben **
1530 REM ********************
1540 IF tim=0 THEN GOTO 1670
1550 IF tim=1 THEN GOTO 1580
1560 DATA 478,15,,5,358,10,,20,358,20,,3
,301,10,,20,301,20,,5,239,35,,10,301,40
1570 RESTORE 1560:FOR y=1 TO 13:READ i,l
:SOUND 1,i,l,15:NEXT
1580 PEN 2:LOCATE 18,typ+1:PRINT USING "
#####";win:LOCATE 17,1:PRINT USING "#";t
im:FOR i=1 TO 1250:NEXT
1590 ti=tim-1
1600 FOR y=1 TO tim
1610 wi=win
1620 FOR p=1 TO 100:NEXT
1630 FOR i=bet TO wi STEP bet
1640 wi=wi-bet:sc=sc+bet:SOUND 1,45,3,15
1650 PEN 2:LOCATE 18,typ+1:PRINT USING "
#####";wi:LOCATE 17,1:PRINT USING "#";ti
:PEN 1:LOCATE 30,4:PRINT USING "######";
sc
1660 NEXT:ti=ti-1:NEXT
1670 PEN 3:LOCATE 2,typ+1:PRINT a$;
1680 IF typ=0 THEN 1700
1690 LOCATE 18,typ+1:PRINT USING "#####"
;k(typ)
1700 LOCATE 17,1:PRINT CHR$(154);:tim=0:
typ=0:a$=""
1710 RETURN
1720 rr=0:CLS#7:PEN#7,1:PRINT#7,"Take Wi
n (X), Double "CHR$(242)" or Quadruple "
CHR$(243);
1730 IF JOY(0)<4 AND INKEY(8)<>0 AND INK
EY(1)<>0 AND INKEY(9)<>0 THEN 1730
1740 IF JOY(0)=16 OR INKEY(9)=0 THEN tim
=1:GOSUB 1540:GOTO 1780
1750 IF JOY(0)=4 OR INKEY(8)=0 THEN rr=2
:GOTO 1770
1760 IF JOY(0)=8 OR INKEY(1)=0 THEN rr=4
:GOTO 1770
1770 GOSUB 1820
1780 RETURN
1790 REM ************
1800 REM ** Risiko **
1810 REM ************
1820 CLS#6:CLS#3:PEN#3,3:SOUND 2,240,10,
13,0,0,1 :PRINT#3,STRING$(63,127);
1830 FOR i=1 TO 33 STEP 8:LOCATE i,22:PR
INT hd$:NEXT
1840 RESTORE 1850:FOR i=1 TO 7:READ t,l:
SOUND 1,t,l,15:NEXT
1850 DATA 638,10,,20,638,15,,5,638,30,,2
,758,60
1860 IF rr=2 THEN GOSUB 1920
1870 IF rr=4 THEN GOSUB 2030
1880 RETURN
1890 REM **************
1900 REM ** Risiko I **
1910 REM **************
1920 CLS#7:PEN#7,2:PRINT#7,"
R  I  S  K   I"
1930 CLS#2:PEN#2,0:PRINT#2,STRING$(63,14
3);:CLS#4:PEN#4,2:PRINT#4,STRING$(63,143
);
1940 IF (JOY(0)<4 OR JOY(0)>8) AND INKEY
(8)<>0 AND INKEY(1)<>0 THEN 1940
1950 IF JOY(0)=4 OR INKEY(8)=0 THEN rp=0
:GOTO 1970
1960 IF JOY(0)=8 OR INKEY(1)=0 THEN rp=2
:GOTO 1970
1970 GOSUB 2210
1980 IF rp=p THEN tim=2 ELSE tim=0:win=0
:FOR i=80 TO 1500 STEP 40:SOUND 1,i,5,15
:NEXT
1990 RETURN
2000 REM ***************
2010 REM ** Risiko II **
2020 REM ***************
2030 CLS#7:PEN#7,2:PRINT#7,"
R  I  S  K   II":as=128
2040 ran=13:ra=0:FOR w=1 TO 5:ra=ra+1:ON
 w GOSUB 2270,2270,2170,2270,2270:NEXT
2050 PEN 2:l=17:rb=3:as=65
```

```
2060 IF JOY(0)=16 OR INKEY(9)=0 THEN GOT
O 2130
2070 LOCATE 1,22:PRINT hs$
2080 IF l>25 THEN l=33:GOTO 2100
2090 IF JOY(0)=8 OR INKEY(1)=0 THEN FOR
f=1 TO 100:NEXT:LOCATE 1,22:PRINT hd$;:l
=l+8:rb=rb+1:GOTO 2060
2100 IF l<8 THEN l=1:GOTO 2060
2110 IF JOY(0)=4 OR INKEY(8)=0 THEN FOR
f=1 TO 100:NEXT:LOCATE 1,22:PRINT hd$;:l
=l-8:rb=rb-1:GOTO 2060
2120 GOTO 2060
2130 IF l=17 THEN 2060
2140 IF rb>3 THEN rb=rb-1
2150 GOSUB 2210
2160 IF rb=bb(3)THEN tim=4 ELSE tim=0:wi
n=0:FOR i=80 TO 1500 STEP 40:SOUND 1,i,5
,15:NEXT:RETURN
2170 ra=ra-1:RETURN
2180 REM ***********************
2190 REM ** Karten auswaehlen **
2200 REM ***********************
2210 RANDOMIZE TIME
2220 ra=INT(RND(6)*5)
2230 IF ra<1 OR ra>4 THEN GOTO 2220
2240 ran=INT(RND(6)*14)
2250 IF ran<ram OR ran>13 THEN GOTO 2240
2260 w=3:GOSUB 2270:RETURN
2270 bb(w)=ra:aa(w)=ran:s=160+bb(w):z=18
9+bb(w)
2280 IF (bb(w)/2)-INT(bb(w)/2)=0 THEN p=
2 ELSE p=0
2290 CLS#w:PEN#w,p:SOUND 2,240,10,13,0,0
,1
2300 ON ran GOSUB 2600,2600,2620,2620,26
20,2640,2640,2420,2430,2660,2670,2680,26
90
2310 RETURN
2320 REM *************
2330 REM ** Matrixe **
2340 REM *************
2350 REM ** 2,3 **
2360 PRINT#w,CHR$(v)SPACE$(5)CHR$(v)SPAC
E$(10)CHR$(s),SPACE$(10)CHR$(y),SPACE$(1
0)CHR$(s)SPACE$(4),CHR$(v)SPACE$(5)CHR$(
v);:RETURN
2370 REM ** 4,5,6 **
2380 PRINT#w,CHR$(v)SPACE$(5)CHR$(v)SPAC
E$(8)CHR$(s)SPACE$(3)CHR$(s)SPACE$(8)CHR
$(y)CHR$(128)CHR$(x)CHR$(128)CHR$(y)SPAC
E$(8)CHR$(s)SPACE$(3)CHR$(s)SPACE$(7)CHR
$(v)SPACE$(5)CHR$(v);:RETURN
2390 REM ** 7,8 **
2400 PRINT#w,CHR$(v)SPACE$(5)CHR$(v)SPAC
E$(8)CHR$(s)SPACE$(3)CHR$(s)SPACE$(3)CHR
$(s),CHR$(128)CHR$(s)SPACE$(3)CHR$(s),SP
ACE$(3)CHR$(y),CHR$(128)CHR$(s)SPACE$(3)
CHR$(s)SPACE$(7)CHR$(v)SPACE$(5)CHR$(v);
:RETURN
2410 REM ** 9 **
2420 PRINT#w,CHR$(57);SPACE$(5);CHR$(57)
;CHR$(128);CHR$(s);SPACE$(3);CHR$(s),SPA
CE$(8);CHR$(s);SPACE$(3);CHR$(s),SPACE$(
3);CHR$(s),CHR$(128);CHR$(s);SPACE$(3);C
HR$(s),SPACE$(8);CHR$(s);SPACE$(3);CHR$(
s),CHR$(57);SPACE$(5);CHR$(57);:RETURN
2430 REM ** 10 **
2440 PRINT#w,te$;SPACE$(3);te$;CHR$(128)
CHR$(s)SPACE$(3)CHR$(s),SPACE$(3)CHR$(s)
,CHR$(128)CHR$(s)SPACE$(3)CHR$(s),SPACE$
(8)CHR$(s)SPACE$(3)CHR$(s),SPACE$(3)CHR$
(s),CHR$(128)CHR$(s)SPACE$(3)CHR$(s),te$
;SPACE$(3);te$;:RETURN
2450 REM ** A **
2460 PRINT#w,CHR$(as)SPACE$(5)CHR$(as),S
PACE$(2)CHR$(a(1))CHR$(a(2))CHR$(a(3)),S
PACE$(2)CHR$(a(4))CHR$(a(5))CHR$(a(6)),S
PACE$(2)CHR$(a(7))CHR$(a(8))CHR$(a(9))SP
ACE$(14)CHR$(as)SPACE$(5)CHR$(as);:RETUR
N
2470 REM ** B,D,K **
2480 PRINT#w,CHR$(b)CHR$(128)CHR$(s)CHR$
(128)CHR$(z)CHR$(128)CHR$(b):PRINT#w,CHR
$(z)CHR$(128)CHR$(f(1))CHR$(f(2))CHR$(f(
3))CHR$(128)CHR$(s)SPACE$(2)CHR$(f(4))CH
R$(f(5))CHR$(f(6)),CHR$(z)CHR$(128)CHR$(
f(7))CHR$(f(8));
2490 PRINT#w,CHR$(f(9))CHR$(128)CHR$(s)S
PACE$(2)CHR$(f(10))CHR$(f(11))CHR$(f(12)
),CHR$(z)CHR$(128)CHR$(f(13))CHR$(f(14))
CHR$(f(15))CHR$(128)CHR$(s):PRINT#w,CHR$
(b)CHR$(128)CHR$(s)CHR$(128)CHR$(z)CHR$(
128)CHR$(b);:RETURN
2500 REM ****************************
2510 REM ** Variabelen fuer Matrixe **
2520 REM ****************************
2530 DATA 165,166,167,168,143,169,170,17
1,172
2540 DATA 173,174,175,176,177,178,128,17
9,128
2550 DATA 128,180,128,181,182,183,184,18
5,172
2560 DATA 128,186,128,187,188,189,128,17
9,128
2570 DATA 224,225,226,227,228,229,230,23
1,232,233,234,235,236,237,238
2580 DATA 209,210,211,212,213,214,215,21
6,217,218,219,220,221,222,223
2590 DATA 194,195,196,197,198,199,200,20
1,202,203,204,205,206,207,208
2600 IF aa(w)=1 THEN v=50:y=128 ELSE v=5
1:y=s
2610 GOSUB 2360:RETURN
2620 IF aa(w)=3 THEN v=52:x=128:y=128 EL
SE IF aa(w)=4 THEN v=53:x=s:y=128 ELSE v
=54:y=s:x=128
2630 GOSUB 2380:RETURN
2640 IF aa(w)=6 THEN v=55:y=128 ELSE v=5
6:y=s
2650 GOSUB 2400:RETURN
2660 RESTORE 2570:FOR d=1 TO 15:READ f(d
):NEXT:b=66:GOSUB 2480:RETURN
2670 RESTORE 2580:FOR d=1 TO 15:READ f(d
):NEXT:b=68:GOSUB 2480:RETURN
2680 RESTORE 2590:FOR d=1 TO 15:READ f(d
):NEXT:b=75:GOSUB 2480:RETURN
```

```
2690 ON ra GOSUB 2710,2720,2730,2740
2700 RETURN
2710 RESTORE 2530:FOR d=1 TO 9:READ a(d)
:NEXT:GOSUB 2460:GOTO 2700
2720 RESTORE 2540:FOR d=1 TO 9:READ a(d)
:NEXT:GOSUB 2460:GOTO 2700
2730 RESTORE 2550:FOR d=1 TO 9:READ a(d)
:NEXT:GOSUB 2460:GOTO 2700
2740 RESTORE 2560:FOR d=1 TO 9:READ a(d)
:NEXT:GOSUB 2460:GOTO 2700
2750 REM ******************
2760 REM ** Programmende **
2770 REM ******************
2780 MODE 0:INK 0,0:INK 6,0:INK 7,0:INK
8,0:PEN 6
2790 FOR y=1 TO 13
2800 FOR i=1 TO 10:PRINT CHR$(128);CHR$(
143);:NEXT
2810 FOR i=1 TO 10:PRINT CHR$(143);CHR$(
128);:NEXT
2820 NEXT
2830 INK 0,7,15:INK 6,9,1:INK 7,1,3
2840 WINDOW#7,5,16,11,15:PAPER#7,8:CLS#7
:PEN#7,7:PRINT#7:PRINT#7:PRINT#7," YOU B
USTED"
2850 FOR i=20 TO 1000 STEP 5:SOUND 1,i,5
,15:NEXT:FOR i=1 TO 500:NEXT
2860 MODE 1:INK 0,0:CLS
2870 LOCATE 9,10:PRINT"Do you want a new
 game ?"
2880 LOCATE 18,14:PRINT"[Y/N]"
2890 yn$=INKEY$:IF yn$="" THEN GOTO 2890
2900 IF UPPER$(yn$)="Y" THEN RUN 440
2910 IF UPPER$(yn$)="N" THEN INK 1,24:PE
N 1:CLS:END
2920 GOTO 2890
```

Checksummen

ZEILENNR.:		SUMMEN
10-	15:	44541
20-	25:	47820
30-	35:	46305
40-	45:	23864
50-	55:	63185
60-	65:	27239
70-	75:	43057
80-	100:	24018
110-	120:	64812
130-	140:	76044
150-	160:	16800
170-	180:	21163
190-	200:	21191
210-	220:	46096
230-	240:	32180
250-	260:	79648
270-	280:	38207
290-	300:	14990
310-	320:	114511
330-	340:	106444
350-	360:	126909
370-	380:	32342
390-	400:	59668
410-	420:	37750
430-	440:	29494
450-	460:	45917
470-	480:	56455
490-	500:	127778
510-	520:	59544
530-	540:	68838
550-	560:	16359
570-	580:	27738
590-	600:	56382
610-	620:	56347
630-	640:	31860
650-	660:	116764
670-	680:	7226
690-	700:	19879
710-	720:	46888
730-	740:	71384
750-	760:	53645
770-	780:	29369
790-	800:	11993
810-	820:	13573
830-	840:	47727
850-	860:	94963
870-	880:	31683
890-	900:	13594
910-	920:	79015
930-	940:	36039
950-	960:	51272
970-	980:	77465
990-	1000:	54281
1010-	1020:	49646
1030-	1040:	55452
1050-	1060:	10726
1070-	1080:	27174
1090-	1100:	46780
1110-	1120:	46398
1130-	1140:	22463
1150-	1160:	16862
1170-	1180:	4405
1190-	1200:	16454
1210-	1220:	35274
1230-	1240:	377
1250-	1260:	20676
1270-	1280:	54668
1290-	1300:	79807
1310-	1320:	116940
1330-	1340:	94794
1350-	1360:	92345
1370-	1380:	83255
1390-	1400:	23912
1410-	1420:	29537
1430-	1440:	18556
1450-	1460:	67550
1470-	1480:	68919
1490-	1500:	1216
1510-	1520:	33256
1530-	1540:	25407
1550-	1560:	69409
1570-	1580:	16122
1590-	1600:	25170
1610-	1620:	18424
1630-	1640:	59839
1650-	1660:	35627
1670-	1680:	33749
1690-	1700:	55539
1710-	1720:	22125
1730-	1740:	23751
1750-	1760:	104184
1770-	1780:	597
1790-	1800:	11489
1810-	1820:	5675
1830-	1840:	54124
1850-	1860:	50314
1870-	1880:	13882
1890-	1900:	13908
1910-	1920:	26353
1930-	1940:	41355
1950-	1960:	100222
1970-	1980:	34360
1990-	2000:	6698
2010-	2020:	15728
2030-	2040:	78716
2050-	2060:	62065
2070-	2080:	31170
2090-	2100:	40615
2110-	2120:	23437
2130-	2140:	38358
2150-	2160:	3647
2170-	2180:	26619
2190-	2200:	40247
2210-	2220:	15822
2230-	2240:	48834
2250-	2260:	62909
2270-	2280:	103016
2290-	2300:	59797
2310-	2320:	5312
2330-	2340:	13497
2350-	2360:	53892
2370-	2380:	6206
2390-	2400:	54808
2410-	2420:	7452
2430-	2440:	4351
2450-	2460:	28775
2470-	2480:	67687
2490-	2500:	46588
2510-	2520:	61132
2530-	2540:	70709
2550-	2560:	70946
2570-	2580:	55235
2590-	2600:	77153
2610-	2620:	9358
2630-	2640:	52025
2650-	2660:	24858
2670-	2680:	46116
2690-	2700:	21364
2710-	2720:	130780
2730-	2740:	129921
2750-	2760:	25426
2770-	2780:	29608
2790-	2800:	45208
2810-	2820:	38966
2830-	2840:	34207
2850-	2860:	35424
2870-	2880:	51690
2890-	2900:	52393
2910-	2920:	34567

GESAMTCHECKSUMME: 25519

Werkstatt:

DISKETTEN-ARCHIV für Schneider CPC 464,664 und 6128

DISKETTEN-ARCHIV ist ein Programm zur Verwaltung Ihrer Programm- und Datendisketten. Mit DISKETTEN-ARCHIV können Sie sich, ohne lästigen Diskettenwechsel, die Inhalte Ihrer Disketten ansehen, oder aber gezielt nach einem bestimmten File suchen. Bis zu 19 Disketten können verwaltet werden.

Nach dem Start des Programms wird Ihnen zunächst das auf die letzte Aktualisierung folgende Datum angezeigt. Haben Sie das Datum bestätigt, bzw. das korrekte Datum eingegeben, gelangen Sie in das Menü von DISKETTEN-ARCHIV. Es stehen fünf Menüpunkte zur Auswahl!

AKTUALISIEREN: Dient zum Auslesen der Directory einer Diskettenseite. Nach dem Auslesen werden Ihnen die Diskettennummer und -Seite angezeigt. Hatte die Diskette eine ungültige oder keine Nummer, so wird eine entsprechende Meldung ausgegeben.

DIREKTORY ANSEHEN: Über diesen Menüpunkt können Sie sich, nachdem Sie die gewünschte Diskettennummer und-Seite eingegeben haben, die Direktory ausgeben lassen.

In der Kopfzeile werden neben Disketten-

nummer und -Seite das Datum der letzten Aktualisierung und das Format der Diskette ausgegeben. (Das sehr selten benutzte IBM-Format wird nicht berücksichtigt.) Anschließend werden alphabetisch geordnet die Filenamen mit der jeweiligen Filelänge ausgegeben. (Das zur Numerierung der Diskette dienende File wird nicht angezeigt.) Abschließend wird die Anzahl der freien Kbytes der Diskettenseite, sowie die Anzahl der Einträge, abzüglich des Files für die Numerierung, angezeigt.
Auf Wunsch kann die Directory auf einem Drucker ausgegeben werden.

EINTRAG SUCHEN: Diese Option dient, wie der Name schon sagt, zum Suchen eines Files. Ist das File vorhanden, gibt DISKETTEN-ARCHIV den Filenamen, Filelänge, Diskettennummer und -Seite, sowie das Datum, an dem das File zuletzt erfaßt wurde, aus. Konnte das File nicht gefunden werden, so wird Ihnen dies mitgeteilt. Neben der Suche nach einem bestimmten File, können Sie sich über diese Option auch Listen von Filetypen (z. B. alle Basicfiles durch Eingabe von .BAS) oder bestimmten Filenamen erstellen lassen. Die Suchroutine ist in Maschinensprache geschrieben, so daß die Suche sehr rasch geschieht.
Bei der Eingabe des Suchbegriffs ist darauf zu achten, daß dieser zwischen 2 und 12 Zeichen lang sein muß. Der Suchbegriff kann in Klein- oder Großbuchstaben eingegeben werden.
Die Ausgabe kann wahlweise auf dem Bildschirm oder einem Drucker erfolgen.

Disketten numerieren: Voraussetzung für das Funktionieren von DISKETTEN-ARCHIV ist, daß sich auf den Disketten ein File befindet, das zur Identifizierung der Diskette dient. Über diesen Menüpunkt wird ein File auf Ihren Disketten angelegt, aus dessen Name sich die Diskettennummer und aus dem Filetyp die Seite der Diskette ergibt. Obwohl das Numerierungsfile nur 1 Byte lang ist, belegt es unter AMSDOS 1 Kilobyte. Sollten Sie der Meinung sein, daß hierdurch der Ihnen auf der Diskette zur Verfügung stehende Speicherplatz zu sehr verringert wird, können Sie natürlich auch ein vorhandenes File so umbenennen, daß es zur Identifizierung der Diskette herangezogen werden kann.

Programm beenden: Bevor DISKETTEN-ARCHIV beendet wird, können Sie entscheiden, ob Sie die Daten speichern möchten oder nicht.
Haben Sie sich dafür entschieden die Daten abzuspeichern, werden Sie aufgefordert die Diskette, auf der sich DISKETTEN-ARCHIV befindet einzulegen. Ist dies geschehen, werden die Daten nach einer kurzen Verzögerung von ca. 2 bis 5 Sekunden, je nach Anzahl der Einträge, abgespeichert.
Innerhalb der kurzen Zeitspanne zwischen der Bestätigung und dem Abspeichern werden die Daten komprimiert, so daß sie immer möglichst wenig Platz auf der Diskette benötigen.

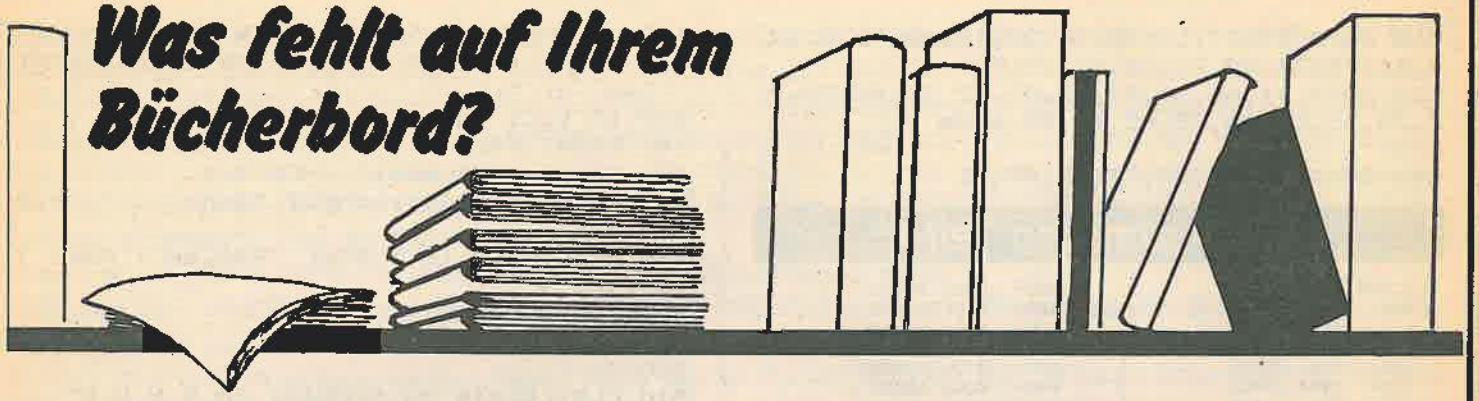
```
100 '*********************************
*******
110 '*** Diskettenarchiv fuer Schneider
CPC ***
120 '*** Basic-Lader fuer Maschinenrouti
nen ***
130 '*********************************
*******
140 '
150 MODE 2:LOCATE 10,10:PRINT "Bitte war
ten..."
160 MEMORY 7994:adr=&A410
170 WHILE a$<>"ende"
180 READ a$ : a=VAL("&"+a$)
190 POKE adr,a : sum=sum+a : adr=adr+1
200 WEND
210 RESTORE 540
220 FOR i=0 TO 6
230 READ a$ : a=VAL("&"+a$)
240 POKE 7995+i,a : sum=sum+a
250 NEXT
260 CLS:LOCATE 10,10
270 IF sum<>44314 THEN PRINT "Fehler in
Data's !":END
280 '
290 PRINT "Bitte legen Sie Ihre 'Archiv'
-Diskette ein und"
300 LOCATE 11,12
310 PRINT "und druecken Sie anschliessen
d eine Taste !"
320 CALL &BB18
330 SAVE "ARCHIV",B,&A410,&167:SAVE "ARC
HIV.DAT",B,7995,7
340
350 DATA 21,5A,BB,7E,36,C9,F5,E5,11,00,9
C,CD,9B,BC,E1,F1,77,C9,21,0E
360 DATA 9C,ED,5B,D7,A4,FD,21,D9,A4,FD,3
6,00,00,3E,FF,BE,20,1B,23,FD
370 DATA 34,00,01,08,00,ED,B0,3E,2E,12,1
3,01,03,00,ED,B0,7E,CB,FF,12
380 DATA 23,13,23,18,E0,3E,FE,12,C9,06,0
B,11,00,C0,21,DC,A4,CD,77,BC
390 DATA C5,21,3B,1F,E5,CD,83,BC,CD,7A,B
C,E1,C1,09,2B,E5,01,3E,03,3E
400 DATA FF,ED,B9,23,22,DA,A4,11,41,1F,A
1,ED,52,E1,D8,E5,2A,DA,A4,E5
410 DATA 23,5E,23,56,C1,E1,A1,E5,ED,42,1
9,E5,C1,E1,7E,02,36,00,FE,FF
420 DATA 0B,2B,20,F6,2A,DA,A4,C3,6E,A4,C
9,3E,FE,21,40,1F,01,74,9A,ED
430 DATA B1,22,DA,A4,3E,FF,01,74,9A,ED,B
1,2B,11,74,9A,E5,A1,ED,52,E1
440 DATA D0,ED,5B,DA,A4,7E,12,36,00,13,2
3,FE,FE,20,F6,EB,C3,B1,A4,C3
450 DATA 5A,BB,ED,43,41,52,43,48,49,56,2
0,2E,44,41,54,21,40,1F,36,00
460 DATA 11,41,1F,01,00,7D,ED,B0,C3,55,A
4,CD,A7,A4,06,0B,21,DC,A4,11
470 DATA 00,C0,CD,8C,BC,3E,02,11,3B,1F,2
A,DA,A4,A1,ED,52,EB,E5,C1,CD
480 DATA 9B,BC,C3,8F,BC,2A,D7,A4,36,00,E
5,D1,13,01,2C,03,ED,B0,C3,22
490 DATA A4,FD,21,D9,A4,AF,32,77,A5,2A,7
A,A5,3A,7D,A5,23,BE,28,0B,EB
500 DATA 21,FF,9B,A7,ED,52,D8,EB,18,F1,2
2,78,A5,3A,7C,A5,47,05,11,7D
```

```
510 DATA A5,23,13,1A,BE,20,1B,10,F8,32,7
7,A5,FD,34,00,23,CB,7E,28,FB
520 DATA 11,0C,00,A7,ED,52,22,78,A5,23,2
2,7A,A5,C9,2A,78,A5,18,BD,ende
530 '
540 DATA 01,01,56,FF,FF,FF,FE
```

Checksummen Teil 1:

ZEILENNR.:	SUMMEN
100- 110:	53382
120- 130:	61924
140- 150:	35761
160- 170:	20371
180- 190:	77926
200- 210:	624
220- 230:	32807
240- 250:	56750
260- 270:	2008
280- 290:	41526
300- 310:	29310
320- 330:	2491
340- 350:	36751
360- 370:	71237
380- 390:	71318
400- 410:	73353
420- 430:	72455
440- 450:	65331
460- 470:	73240
480- 490:	70412
500- 510:	69277
520- 530:	53580
540- :	14926
GESAMTCHECKSUMME:	2325B

```
100 '*********************************
**********
110 '***
       ***
120 '***  Disketten-Archiv fuer Schneide
r CPC  ***
130 '***
       ***
140 '***  (C) Wolfgang Boenigk   Osnabru
eck    ***
150 '***
       ***
160 '*********************************
**********
170 '
180 REM Initialisierung
190 '
200 MODE 2:MEMORY &A000:WIDTH 80:KEY DEF
 66,0,1,1,1
210 LOCATE 20,10:PRINT "Bitte warten..."
220 LOAD "ARCHIV.BIN"
230 MEMORY 7990
240 CALL &A4E7
250 ts$="druecken Sie anschliessend eine
 Taste¯!"
260 uts$="und " + ts$
270 REM ***  Datum  ***
280 tg=PEEK(7995) : tg=tg+1
290 mn=PEEK(7996) : jr=PEEK(7997)
300 IF (tg=29 AND mn=2 AND jr MOD 4) THE
N 350
310 IF tg=30 AND mn=2 AND NOT jr MOD 4 T
HEN 350
320 IF tg=31 AND (mn=4 OR mn=6 OR mn=9 O
R mn=11) THEN 350
330 IF tg<>32 THEN 350
340 tg=1 : mn=mn+1 : IF mn=13 THEN mn=1
: jr=jr+1
350 CLS:LOCATE 10,8 : PRINT "Ist heute d
er ";
360 PRINT USING "##";tg;: PRINT ".";
370 PRINT USING "##";mn;: PRINT ".";
380 PRINT USING "##";jr;
390 PRINT "   J/N ?"
400 a$=INKEY$ : IF a$="" THEN 400
410 IF UPPER$(a$)<>"N" THEN 470
420 PRINT : PRINT :PRINT TAB (10);:INPUT
 "Tag   : ";tg
430 IF tg<1 OR tg > 31 THEN 420
440 PRINT TAB(10);:INPUT "Monat : ";mn
450 IF mn<1 OR mn>12 THEN 440
460 PRINT TAB(10);:INPUT "Jahr  : ";jr:G
OTO 350
470 POKE 7995,tg : POKE 7996,mn : POKE 7
997,jr
480 '
490 REM Menue
500 '
510 CLS:LOCATE 24,5:PRINT "M E N U E"
520 LOCATE 20,8:PRINT "1  Aktualisieren"
530 LOCATE 20,10:PRINT "2  Directory ans
ehen"
540 LOCATE 20,12:PRINT "3  Eintrag suche
n"
550 LOCATE 20,14:PRINT "4  Disketten num
merieren"
560 LOCATE 20,16:PRINT "5  Programm been
den"
570 LOCATE 11,20:PRINT "Geben Sie bitte
die gewuenschte Nummer ein !"
580 a$=INKEY$:IF a$="" THEN 580
590 IF a$<"1" OR a$>"5" THEN 580
600 CLS:ON VAL(a$) GOTO 610,970,1160,162
0,1830
610 '
620 REM AKTUALISIEREN
630 '
640 LOCATE 10,10:PRINT "Soll wirklich ak
tualisiert werden J/N ?"
650 a$=INKEY$:IF a$="" THEN 650 ELSE IF
UPPER$(a$)<>"J" THEN 480
660 CLS:LOCATE 10,7:PRINT "Bitte legen S
ie eine Diskette ein und"
670 LOCATE 10,8:PRINT ts$
680 CALL &BB18 : CALL &A410
690 nr=PEEK(&9C01)-&30
700 IF nr>9 OR nr<1 THEN 920
710 IF nr<>1 THEN 740 ELSE IF PEEK(&9C02
)=32 THEN 740
720 b=PEEK(&9C02)-&30
730 IF b>9 THEN 920 ELSE nr=10+b
740 pg=PEEK(&9C09)-&30
750 IF pg<>1 AND pg<>2 THEN 920
760 LOCATE 10,10:PRINT "Disc :";nr;TAB(2
3);"Seite :";pg
770 ab=((nr-1)*2+pg-1)*830+8000
780 POKE ab,&FF : POKE ab+3,nr
790 POKE ab+4,pg : POKE ab+7,tg
800 POKE ab+8,mn : POKE ab+9,jr
810 abh=INT((ab+10)/256)
820 abl=ab+10-abh*256
830 POKE &A4D7,abl:POKE &A4D8,abh
840 abh=INT(ab/256) : abl=ab-abh*256
850 POKE ab+1,abl : POKE ab+2,abh
860 CALL &A519
870 POKE ab+5,PEEK(&A4D9)
880 POKE ab+6,PEEK(&A89F)
890 LOCATE 10,12:PRINT "Soll eine weiter
e Diskette aktualisiert werden J/N ?"
```



Z80 Maschinensprachekurs für den Schneider CPC 464 / 664 / 6128

von Peter Heiß

Wer seinen Computer richtig einsetzen will, dem reicht früher oder später die Arbeitsgeschwindigkeit eines BASIC-Interpreters nicht mehr aus, so daß er auf die wesentlich schnellere Maschinensprache zurückgreifen muß. Doch spätestens jetzt fangen die Probleme an. Bücher, die die Hard- und Softwareeigenschaften der Schneider-Computer beschreiben, sind zwar schon vorhanden, aber die nützen einem wenig, wenn man nicht weiß, wie der Z80-Prozessor zu programmieren ist. Genau in diese Lücke stößt das Buch „Z80 Maschinensprachekurs" von Peter Heiß. Praktisch jeder Befehl wird anhand von Übungsprogrammen erklärt, was den Vorteil hat, daß der Leser ziemlich schnell zu einem Erfolgserlebnis kommt und die Bedeutung und Wirkung der Befehle fast spielerisch begreift. Nach jedem Kapitel werden Übungsaufgaben gestellt, die mit der Zeit schwieriger werden, aber von jedem leicht gelöst werden können, der das Buch aufmerksam durcharbeitet. Am Ende müßte eigentlich jeder in der Lage sein, eigene Programme zu erstellen, so daß das Lernziel relativ schnell erreicht wird. Zum Anhang des Buches gehört neben einer vollständigen Befehlsübersicht auch das Listing eines Direktassemblers, so daß auch selbstentwickelte Programme leicht eingegeben werden können.

Fazit: Empfehlenswert

Z80 Machinensprachekurs für den Schneider CPC 464 / 664 / 6128
Heise-Verlag, Hannover

DFÜ für Jedermann

Rainer Severin

Datenfernübertragung ist in der Userszene so etwas wie ein Zauberwort. Nicht zuletzt deshalb, weil es ein Kommunikationsmittel darstellt, das ohne Rücksicht auf die Systemvielfalt Gleichgesinnte miteinander verbindet. Ursprünglich war es der Reiz am Hacken (sprich: häcken), also das Eindringen in fremde Datenbanken, was viele User zum Akkustikkoppler greifen ließ. Heute jedoch sind alle wichtigen Datenbanken so kompliziert abgesichert, daß dieser Aspekt kaum noch eine wichtige Bedeutung hat. Dennoch erfreut sich die DFÜ immer größerer Beliebtheit. Elektonischer Briefkasten, Mailboxen mit Kleinanzeigemarkt und Programmübertragung sind nur die populärsten Schlagworte der DFÜ.

Wer sich allerdings als Anfänger an die DFÜ wagt, wird unversehens mit einer Vielzahl von neuen Begriffen, gesetzlichen Bestimmungen und elektronischen Daten konfrontiert.

Rainer Severins Buch setzt genau an dieser Stelle an und führt den Leser in die Feinheiten und Besonderheiten der DFÜ ein. Mit der den Databeckerbüchern selbstverständlichen Gründlichkeit werden alle wichtigen Aspekte der DFÜ behandelt und verständliche Anleitungen für das eigene Arbeiten mit der DFÜ geliefert. Neben einer Reihe von Hardwaretips, Adressen und Mailboxen und einer Softwareübersicht, wird auch ein eigenes Terminalprogramm zum Abtippen angeboten und eine Bauanleitung für eine Autoanswer-Einrichtung präsentiert.
Eine wirklich sehr gelungene Einführung in die Datenfernübertragung, allerdings vermissen wir am Ende des Buchs ein Sachwortregister (in der nächsten Auflage, ja?). (uk)

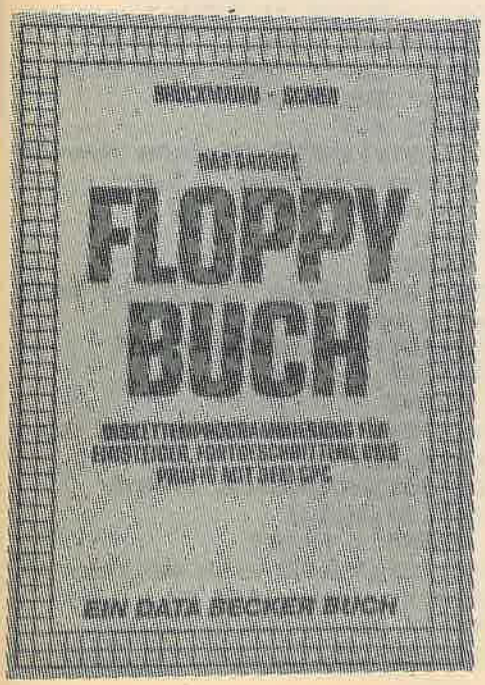
Databecker-Verlag, Düsseldorf
Preis: 39,-- DM

Der Computerdieb

von Emil Zopfi

Das hier vorgestellte Buch stellt kein Fachbuch zum Thema „Computer" im üblichen Sinne dar. Es ist vielmehr ein spannendes Unterhaltungsbuch, das sich mit dem Thema von ganz anderer Seite beschäftigt.
Marcs Traum und seine größte Leidenschaft ist der Besitz eines Computers. Er träumt täglich davon, den Golden Arrow, den besten Computer der Welt zu besitzen. Doch leider hat er kein Geld! Kurz entschlossen beschließt er, während einer Messe seinen Computer zu stehlen. Den Diebstahl plant er wie ein Computerprogramm. Doch wie jedes Programm am Anfang, so hat auch dieser Plan Fehler. Zum Beispiel wußte Marc nicht, daß die Computerfirma eine Sperre gegen Diebstahl in den Computer eingebaut hat, so daß Marc den Computer überhaupt nicht benutzen kann.
In einem Freizeitzentrum trifft Marc Andreas, einen ehemaligen Computerspezialisten. Obwohl der nie wieder etwas mit Computern zu tun haben wollte, reizt es ihn doch, die Sperre zu knacken. So beschließt er zusammen mit Marc gegen den Computer und die Polizei, die Marc mittlerweile auf der Spur ist, zu kämpfen. Wir wollen nicht zu viel über den Inhalt verraten, denn ein wenig Spannung soll schon bleiben. Eine positive Begleiterscheinung des Inhaltes dieses Buches ist die Weitergabe von Fachwissen aus dem Bereich der Programmierung. Der Autor, selbst Computerfachmann, hat mit großem Einfühlungsvermögen und Sachverstand eine Geschichte über die Faszination der Computer geschrieben.

Benzinger Verlag Zürich, Köln ca. 144 S., 16,80 DM, ISBN 3 545 32260 2



Neu für den Schneider:

Floppybuch für den CPC

Typisch für Databeckerbücher ist, daß sie den Leser tief in das Innere der technischen Geheimnisse blicken lassen. So nicht anders bei dem Floppybuch für den Schneider CPC. Neben einer grundlegenden Einführung in das Diskettenbetriebssystem unter CP/M und in das Arbeiten mit AMSDOS werden alle Filetypen genauestens erläutert und in Programmbeispielen demonstriert. Für das fortgeschrittene Programmieren werden die DOS-Vektoren beschrieben. Das komplette DOS-Listing gehört ebenso zum Umfang des Buches wie ein Diskmonitor und ein Diskettenmanager. Da die Programmbeispiele sehr umfangreich sind, wird eine Diskette zum Buch für 29,-- DM angeboten.

Das Buch hat schon beste Dienste in unserer Programmierabteilung geleistet und wird auch von unseren Programmierern wärmstens weiterempfohlen.

Brückmann/Schieb, Das große Floppybuch, Diskettenprogrammierung für Einsteiger, Fortgeschrittene und Profis mit dem CPC
Databecker, Düsseldorf, 49,-- DM

```
900 a$=INKEY$ : IF a$="" THEN 900
910 IF UPPER$(a$)="N" THEN 480 ELSE GOTO
 660
920 REM *** Ungueltige Discnummer ***
930 PRINT:PRINT TAB(10);"Die eingelegte
Diskette hat keine gueltige Nummer !"
940 PRINT : PRINT
950 LOCATE 25,25:PRINT "Weiter auf Taste
ndruck !"
960 CALL &BB18 : GOTO 480
970 '
980 REM Directory ausgeben
990 '
1000 LOCATE 1,5
1010 PRINT TAB(10);:INPUT "Gewuenschte D
isc-Nummer : ";nr
1020 IF nr <1 OR nr>19 THEN 1010 ELSE PR
INT
1030 PRINT TAB(10);:INPUT "Gewuenschte S
eite        : ";pg
1040 IF pg<>1 AND pg<>2 THEN 1030
1050 ab=((nr-1)*2+pg-1)*830+8009
1060 abz=ab
1070 IF PEEK(ab-9)<>&FF THEN PRINT:PRINT
:PRINT TAB(10);"Fuer Disc";nr;" Seite";p
g;"liegen keine Eintraege vor!":GOTO 950
1080 aus=0 : GOSUB 1950
1090 PRINT : PRINT
1100 PRINT TAB(10);"Soll die Directory a
uf den Drucker ausgegeben werden J/N ?"
1110 a$=INKEY$:IF a$="" THEN 1110 ELSE I
F UPPER$(a$)<>"J" THEN 480
1120 PRINT:PRINT TAB(10);"Schalten Sie b
itte Ihren Drucker ein und"
1130 PRINT TAB(10);ts$
1140 CALL &BB18 : aus=8
1150 ab=abz:GOSUB 1950:GOTO 480
1160 '
1170 REM Suchen
1180 '
1190 LOCATE 6,7:L$="":R$=""
1200 PRINT TAB(6);:LINE INPUT "Suchbegri
ff : ";s$
1210 IF s$="" OR LEN(s$)=1 OR LEN(s$)>12
 THEN 1200
1220 s$=UPPER$(s$)
1230 IF LEN(s$)<5 OR LEN(s$)=12 THEN 131
0
1240 IF MID$(s$,LEN(s$)-3,1)<>"." THEN 1
310
1250 REM Blanks einfuegen
1260 l$=LEFT$(s$,LEN(s$)-4) : r$=RIGHT$(
s$,4)
1270 FOR i=LEN(l$) TO 7
1280 l$=l$+" "
1290 NEXT i
1300 s$=l$+r$
1310 POKE &A57C,LEN(s$) : POKE &A4D9,0
1320 POKE &A57A,&40 : POKE &A57B,&1F
1330 FOR i=1 TO LEN(s$)
1340 POKE &A57C+i,ASC(MID$(s$,i,1))
1350 NEXT i
1360 PRINT:PRINT:PRINT TAB(6);"Ausgabe a
uf >B<ildschirm oder >D<rucker ?"
1370 a$=INKEY$:IF a$="" THEN 1370
1380 IF UPPER$(a$)="D" THEN aus=8 ELSE a
us=0
1390 IF aus=8 THEN PRINT:PRINT TAB(6);"S
chalten Sie bitte Ihren Drucker ein und
":PRINT TAB(6);ts$:CALL &BB18
1400 CLS:LOCATE 1,5
1410 CALL &A529
1420 IF PEEK(&A4D9) THEN 1460
1430 LOCATE 20,7:PRINT s$;" ist nicht vo
rhanden !"
1440 LOCATE 25,25:PRINT "Weiter auf Tast
endruck !"
1450 CALL &BB18 : GOTO 480
1460 IF PEEK(&A577)=0 THEN PRINT:GOTO 14
40
1470 ad=PEEK(&A578)+PEEK(&A579)*256
1480 gr=INT((ad-8000)/830)*830+8000
1490 PRINT#aus,TAB(5);
1500 FOR i=ad TO ad+11
1510 PRINT#aus,CHR$(PEEK(i));
1520 NEXT i
1530 PRINT#aus,TAB(17);USING "####";PEEK
(ad+12)-&80;
1540 PRINT#aus,"K";TAB(28);"Disc : ";USI
NG "##";PEEK(gr+3);
1550 PRINT#aus,TAB(42);"Seite : ";USING
"##";PEEK(gr+4);
1560 PRINT#aus,TAB(58);"Stand : ";USING
"##";PEEK(gr+7);
1570 PRINT#aus,".";USING "##";PEEK(gr+8)
;
1580 PRINT#aus,".";USING "##";PEEK(gr+9)
1590 adh=INT((ad+12)/256):adl=ad+12-adh*
256
1600 POKE &A57A,adl:POKE &A57B,adh
1610 GOTO 1410
1620 '
1630 REM Disketten nummerieren
1640 '
1650 LOCATE 10,10:PRINT "Sollen wirklich
 Disketten nummeriert werden  J/N ?"
1660 a$=INKEY$:IF a$="" THEN 1660 ELSE I
F UPPER$(a$)<>"J" THEN 480
1670 CLS : LOCATE 1,7
1680 PRINT TAB(10);:INPUT "Welche Nummer
 soll die Diskette erhalten ? ",nr
1690 IF nr<1 OR nr>19 THEN 1680
1700 nr$=STR$(nr)
1710 PRINT : PRINT
1720 PRINT TAB(10);"Legen Sie bitte Seit
e 1 der Diskette ein"
1730 PRINT TAB(10);uts$:CALL &BB18
1740 SAVE nr$+".1",b,1,1
1750 PRINT : PRINT
1760 PRINT TAB(10);"Legen Sie bitte Seit
e 2 der Diskette ein"
1770 PRINT TAB(10);uts$:CALL &BB18
1780 SAVE nr$+".2",b,1,1
1790 PRINT : PRINT : PRINT
1800 PRINT TAB(10);"Soll eine weitere Di
skette nummeriert werden J/N ?"
1810 a$=INKEY$:IF a$="" THEN 1810
1820 IF UPPER$(a$)<>"J" THEN 480 ELSE 16
```

```
70
1830 '
1840 REM Programm beenden
1850 '
1860 LOCATE 5,7:PRINT "Soll das Programm
 wirklich beendet werden J/N ?"
1870 a$=INKEY$:IF a$="" THEN 1870 ELSE I
F UPPER$(a$)<>"J" THEN 480
1880 LOCATE 5,10:PRINT "Sollen die Daten
 gespeichert werden  J/N ?"
1890 a$=INKEY$:IF a$="" THEN 1890 ELSE I
F UPPER$(a$)="N" THEN 1930
1900 LOCATE 5,10:PRINT "Bitte legen Sie
die Programmdiskette ein  "
1910 LOCATE 5,11:PRINT uts$
1920 CALL &BB18 : CALL &A4F7
1930 CALL 0
1940 '
1950 REM Unterprogramm Directory ausgebe
n
1960 '
1970 CLS:PRINT#aus," Disc    : ";USING "#
#";PEEK(ab-6);
1980 PRINT#aus,TAB(18);"Seite : ";USING
"##";PEEK(ab-5);
1990 PRINT#aus,TAB(38);"FORMAT : ";
2000 IF PEEK(ab-3)=&41 THEN PRINT#aus,"C
PM"; ELSE PRINT#aus,"DATA";
2010 IF PEEK(ab-3)=&41 THEN fr=169 ELSE
fr=178
2020 PRINT#aus,TAB(58);"Stand : ";
2030 PRINT#aus,USING "##";PEEK(ab-2);:PR
INT#aus,".";
2040 PRINT#aus,USING "##";PEEK(ab-1);:PR
INT#aus,".";
2050 PRINT#aus,USING "##";PEEK(ab):PRINT
#aus
2060 FOR i=1 TO PEEK(ab-4)
2070 FOR b=1 TO 12
2080 PRINT#aus,CHR$(PEEK(ab+b));
2090 NEXT b
2100 PRINT#aus,USING "####";PEEK(ab+13)-
&80;
2110 PRINT#aus,"K    ";
2120 ab=ab+13
2130 NEXT i
2140 PRINT#aus:PRINT#aus
2150 FOR i=1 TO PEEK(abz-4)
2160 fr=fr-(PEEK(abz+13*i)-&80)
2170 NEXT i
2180 PRINT#aus,fr-1;"K  free";TAB(20);PE
EK(abz-4);" Eintraege"
2190 IF aus=8 THEN PRINT#8:PRINT#8:PRINT
#8
2200 RETURN
```

```
CHECK V1:PRUEFSUMMEN:

ZEILENNR.:  SUMMEN
  100-  110: 54320
  120-  130: 24138
  140-  150: 15812
  160-  170: 48020
  180-  190: 16695
  200-  210: 47354
  220-  230:  7375
  240-  250: 31970
  260-  270: 27172
  280-  290: 68213
  300-  310:123242
  320-  330: 19619
  340-  350: 57031
  360-  370: 30591
  380-  390: 11101
  400-  410: 40580
  420-  430: 62683
  440-  450: 51646
  460-  470: 75233
  480-  490:  3176
  500-  510: 11262
  520-  530: 74941
  540-  550: 92276
  560-  570:104390
  580-  590: 45913
  600-  610: 29641
  620-  630:  9390
  640-  650: 63115
  660-  670: 46612
  680-  690: 17764
  700-  710: 82092
  720-  730: 50394
  740-  750: 39947
  760-  770: 67138
  780-  790: 74411
  800-  810: 66639
  820-  830: 59499
  840-  850: 68552
  860-  870: 16605
  880-  890: 57581
  900-  910: 53006
  920-  930: 79943
  940-  950: 54568
  960-  970:  6109
  980-  990: 21345
 1000- 1010: 10754
 1020- 1030:100511
 1040- 1050: 83122
 1060- 1070: 14201
 1080- 1090: 10198
 1100- 1110: 80278
 1120- 1130: 56655
 1140- 1150: 34084
 1160- 1170:  4005
 1180- 1190: 16215
 1200- 1210: 45704
 1220- 1230: 44523
 1240- 1250: 51766
 1260- 1270: 16961
 1280- 1290:  9153
 1300- 1310: 31561
 1320- 1330: 26925
 1340- 1350: 30262
 1360- 1370: 76147
 1380- 1390: 72210
 1400- 1410:  2323
 1420- 1430: 67562
 1440- 1450: 59143
 1460- 1470: 49548
 1480- 1490: 54362
 1500- 1510: 41902
 1520- 1530: 66814
 1540- 1550: 29536
 1560- 1570: 51540
 1580- 1590: 87765
 1600- 1610: 27348
 1620- 1630: 28699
 1640- 1650: 27964
 1660- 1670: 22156
 1680- 1690: 51869
 1700- 1710: 12144
 1720- 1730: 52864
 1740- 1750: 14248
 1760- 1770: 51637
 1780- 1790: 15985
 1800- 1810: 35631
 1820- 1830: 25051
 1840- 1850: 16946
 1860- 1870: 22082
 1880- 1890: 60336
 1900- 1910: 60838
 1920- 1930:  8009
 1940- 1950: 59053
 1960- 1970: 65555
 1980- 1990: 32103
 2000- 2010: 48698
 2020- 2030: 87658
 2040- 2050:109738
 2060- 2070: 28179
 2080- 2090: 34298
 2100- 2110: 55456
 2120- 2130: 11560
 2140- 2150: 38452
 2160- 2170: 62247
 2180- 2190: 29512
 2200-     :   201
-----------------------
GESAMTCHECKSUMME:  43630
```

CAVE-WALKER

Schneider CPC 464

Ziel des Spiels ist es, „Jonni" den Archäologen durch neun Höhlen zu steuern und in der letzten ein Schwert einzusammeln. „Jonni" wird dabei von „Steino", dem Urzeitmensch, einem fliegenden Totenkopf, herabfallenden Steinen und brodelnden Lavaquellen behindert. Falls eine Perle in der Höhle liegt, sollte sie eingesammelt werden, denn das bringt Punkte und die werden im weiteren Verlauf des Spieles benötigt. Wenn nämlich in der letzten Höhle der Score unter 800 liegt, findet sich „Jonni" plötzlich in der 5. Höhle wieder. Vorsicht! Verschiedene Höhlen sind nicht zu bewältigen. Dann muß „Jonni" zurück gehen.

```
10 'SYMBOLBESTIMMUNG
20 SYMBOL AFTER 0
30 SYMBOL 200,127,119,187,187,183,127,54
,119
40 SYMBOL 201,54,27,51,51,99,99
50 SYMBOL 202,0,0,0,0,0,142,95,31
60 SYMBOL 203,100,62,79,254,56,124,24,24
70 SYMBOL 204,30,119,187,187,183,127,54,
119
80 SYMBOL 229,0,32,223,255,199,64
90 SYMBOL 230,0,0,0,0,0,0,112,250
100 SYMBOL 231,124,242,127,28,30,24,24,6
```

```
3
110 SYMBOL 232,127,223,111,222,222,222,2
23,15
120 SYMBOL 233,27,27,27,27,27
130 SYMBOL 234,56,84,254,238,124,40,56,1
6
140 SYMBOL 235,16,56,16,84,254,84,16,16
150 SYMBOL 236,16,16,16,16,16,16,16,16
160 SYMBOL 237,56,124,78,87,9,24,24
170 SYMBOL 238,60,118,239,239,246,124,56
,108
180 SYMBOL 239,108,198,195,195,195,231,0
,0
190 SYMBOL 240,0,0,128,192,96,48,96,192
200 SYMBOL 241,255,255,127,127,63,15,31,
7
210 SYMBOL 242,255,255,254,254,252,240,2
48,224
220 SYMBOL 243,255,255,126,60,48,56,48,1
6
230 SYMBOL 244,0,0,0,0,0,60,126,255
240 SYMBOL 245,31,5,3,30,124,80
250 SYMBOL 246,16,56,108,62,56,8,8,0
260 SYMBOL 248,171,182,236,172,120,120,4
8,48
270 SYMBOL 249,138,149,75,83,147,171,215
,175
280 SYMBOL 250,8,28,62,126,126,255,255,2
55
290 SYMBOL 251,108,68,130,130,68,68,40,2
0
300 SYMBOL 252,0,0,0,0,16,16,56,56
310 SYMBOL 253,28,56,16,56,120,56,64,16
320 'TITELBILD
330 MODE 1:INK 0,0:BORDER 1:INK 1,18:INK
 2,13:INK 3,2:PAPER 0:PEN 1:CLS
340 PEN 1:A$=CHR$(234):LOCATE 1,1:PRINT
A$;A$;A$;"  ";A$;"  ";A$;" ";A$;" ";A$;A
$;A$:
350 A$=CHR$(234):LOCATE 1,2:PRINT A$;"
 ";A$;" ";A$;" ";A$;" ";A$;" ";A$
360 A$=CHR$(234):LOCATE 1,3:PRINT A$;"
 ";A$;" ";A$;" ";A$;" ";A$;" ";A$;A$
370 A$=CHR$(234):LOCATE 1,4:PRINT A$;"
 ";A$;A$;A$;" ";A$;" ";A$;" ";A$
380 A$=CHR$(234):LOCATE 1,5:PRINT A$;A$;
A$;" ";A$;" ";A$;"  ";A$;"  ";A$;A$;A$
390 ASCII$="-WALKER"
400 POKE &B338,219
410 ORIGIN 260,340
420 FOR K=1 TO LEN (ASCII$)
430 ASCII=ASC(MID$(ASCII$,K,1))
440 CHARADR=HIMEM+ASCII*8+8
450 FOR I=1 TO 8
460 A=PEEK(CHARADR)
470 BINAER$=BIN$(A,8)
480 FOR J=1 TO 8
490 SEGMENT$=MID$(BINAER$,J,1)
500 IF SEGMENT$="0" THEN 550
510 PLOT X+J*4,I*4
520 PLOT X+2+J*4,I*4
530 PLOT X+J*4,I*4+5
540 PLOT X+2+J*4,I*4+2
550 NEXT J:CHARADR=CHARADR-1:NEXT I
560 X=X+40
570 NEXT K
580 PEN 2:FOR A=1 TO 39:LOCATE A,7:FOR K
=1 TO 60::NEXT K:SOUND 1,25,1,7:PRINT CH
R$(234):NEXT
590 PEN 2:LOCATE 20,9:PRINT "MENUE:":LOC
ATE 1,12:PRINT "Erklaerung_____________
___________ (1)":PRINT:PRINT "Start_____
___________________________ (2)"
600 PEN 3:LOCATE 1,19:PRINT STRING$(39,"
*"):LOCATE 1,25:PRINT STRING$(39,"*"):FO
R A=20 TO 24:LOCATE 1,A:PRINT "*":LOCATE
 39,A:PRINT "*":NEXT:PEN 1:LOCATE 3,21:P
RINT "(C) 1986 By Michael Bussmann":LOCA
TE 8,23:PRINT "and    Lutz Klein.":PEN 2
610 ENV 1,2,7,1,1,-1,1,12,-1,10:ENV 2,1,
15,1,15,-1,10:ENV 3,11,1,1,11,-1,6
620 READ TONE:IF TONE=1 THEN 670
630 IF TONE=80 THEN PAUSE=60 ELSE PAUSE=
30
640 SOUND 2,TONE,PAUSE,0,1,1:SOUND 4,TON
E*2,PAUSE,0,2,1:SOUND 1,TONE/2,PAUSE,0,3
,1
650 GOTO 620
660 DATA 159,119,119,119,127,119,106,106
,106,119,106,95,95,95,106,95,89,89,89,95
,89,80,80,95,106,119,106,106,106,119,127
,119,119,119,119,1
670 LOCATE 1,17:PRINT "Ihre Wahl bitte..
...":
680 O$=INKEY$:IF O$="" THEN 680
690 IF O$<>"1" AND O$<>"2" THEN 680
700 IF O$="2" THEN leben=4:GOTO 830
710 MODE 1:BORDER 0:INK 0,0:INK 1,2:INK
2,9:INK 3,6,2
720 PEN 1:FOR A=1 TO 40:LOCATE A,1:PRINT
 CHR$(241);:SOUND 1,A*15,1,0,1,1:LOCATE
A,25:PRINT CHR$(241);:NEXT A:FOR A=1 TO
25:SOUND 4,A*10,1,0,1,1:LOCATE 1,A:PRINT
 CHR$(241);:LOCATE 40,A:PRINT CHR$(241);
:NEXT A
730 :POKE &B1C8,0:POKE &B1CF,&CC:POKE &B
1D0,&33:
740 BORDER 3:LOCATE 5,3:PEN 3:PRINT"CAVE
-WALKER":PEN 2:POKE &B1C8,1:POKE &B1CF,&
88:POKE &B1D0,&44:POKE &B1D1,&22:POKE &B
1D2,&11:
750 PEN 2:LOCATE 8,4:PRINT "------------
------------"
760 LOCATE 3,6:PRINT"Das Spiel laeuft in
 einer Hoehle ab.":LOCATE 3,8:PRINT"Sie
muessen versuchen,die in der":LOCATE 3,1
0:PRINT"Hoehle liegenden Perlen zu samme
ln.":LOCATE 3,12:PRINT"Dabei behindern S
ie Lavaquellen,"
770 LOCATE 3,16:PRINT"herrabfallende Fel
sbrocken.":LOCATE 3,18:PRINT "Um aus der
 Hoehle zu kommen,muessen":LOCATE 3,20:P
RINT "Sie ein Zauberschwert einsammeln."
780 LOCATE 3,14:PRINT "Hoehlenbewohner,g
rosse Steine und"
790 LOCATE 3,23:PEN 3:PRINT "Bitte druec
ken Sie [TAB]!"
800 WHILE INKEY(68)<>0:WEND
```

```
810 BORDER 1:RUN
820 'MAINROUTINE
830 OT=TIME:farb=INT(RND*25)+1:IF farb=3
 OR farb=1 THEN 830 ELSE INK 1,3:INK 2,1
:INK 3,6,26:SPEED INK 10,5:MODE 1:INK 0,
farb:BORDER 0:
840 IF SCHWERT=1 THEN LOCATE SCHWERTP,10
:PRINT CHR$(235):LOCATE SCHWERTP,11:PRIN
T CHR$(236)
850 FOR A=1 TO 40:FOR B=1 TO 3:PAPER 1:L
OCATE A,B:PRINT " ";:NEXT B,A
860 ZUF=INT(RND*3):IF ZUF=0 THEN A$=CHR$
(241):O$=CHR$(242) ELSE IF ZUF=1 THEN A$
=CHR$(243):O$=CHR$(241) ELSE IF ZUF=2 TH
EN A$=CHR$(242):O$=CHR$(243)
870 PAPER 0:PEN 1:FOR A=1 TO 40 STEP 2
880 LOCATE a,4:PRINT a$;o$;
890 NEXT a:
900 PAPER 1:FOR a=1 TO 40:FOR b=25 TO 18
 STEP -1:LOCATE a,b:PRINT " ";:NEXT b,a:
PAPER 0:PEN 1:FOR a=1 TO 40:LOCATE a,17:
PRINT CHR$(244);:NEXT a
910 x=INT(RND*28)+7:X1=INT(RND*28)+7:XM=
1:YM=14:IF X=X1 THEN 910
920 PERLE=INT(RND*37)+1:IF PERLE=X OR PE
RLE=X+1 OR PERLE=X1 OR PERLE=X1+1 THEN 9
20
```

Korrekturen !!!

The little Puck goes home für C-64 in Ausgabe 6

Die Ladeanweisungen sind mißverständlich erklärt gewesen. So starten Sie den **LITTLE PUCK** richtig: 1. Abtippen des Gesamtprogramms; 2. abspeichern; 3. die angegebenen **Pokebefehle im** Direktmodus eingeben; 4. das Programm neu laden und starten; 5. das Programm mit **LOAD"PUCK[16400]"**. **[Geräteadresse],1** laden und mit **SYS16400** starten.

Tips und Tricks für den VC-20 in Ausgabe 5, S. 22

In der Programmzeile **90** befindet sich **ein Dreher!**
Richtig **lautet sie: 90 POKE A+4609:PEEK(A+4609)+18**

```
930 LOCATE PERLE,17:PRINT CHR$(246)
940 GRUB1=INT(RND*30+7):GRUB=INT(RND*30+
7):LOCATE X,17:PEN 3:PRINT CHR$(244);CHR
$(244):LOCATE X1,17:PRINT CHR$(244);CHR$
(244):PEN 2:LOCATE GRUB1,17:PRINT CHR$(2
50);CHR$(250);CHR$(250):LOCATE GRUB,17:P
RINT CHR$(250);CHR$(250);CHR$(250)
950 PRINT CHR$(22)CHR$(1)
960 PRINT CHR$(22);CHR$(1):LOCATE 1,1:PE
N 2:PRINT "Lives:";LEBEN:LOCATE 20,1:PRI
```

weiter Seite 62

★ Kleinanzeigen ★ Kleinanzeigen ★ Kleinanzeigen ★

Auftrag für Gelegenheitsanzeigen in „Compute mit..."

An „Compute mit ..."
Tronic-Verlag
Postfach
3440 Eschwege

Unter der Rubrik „Kleinanzeigen"-veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

Unser Preis für „Kleinanzeigen":
Private Kleinanzeigen
bis 5 Zeilen nur DM 5,– DM
bis 10 Zeilen nur DM 10,– DM

Jede gewerbliche Gelegenheitsanzeige
bis 10 Zeilen nur 15,– DM
(dürfen nicht unter Chiffre erscheinen)

Eine Veröffentlichung erfolgt **nur** gegen Vorkasse!

Kleinanzeigen jetzt noch preiswerter

Name und Adresse _____

_____

Unterschrift _____ Datum _____

Ich wünsche folgenden Text zu veröffentlichen:

5,–

10,–

Zutreffendes ankreuzen

suche ○ Software suche ○ Hardware ○ Tausch ○ Versch.
biete an ○ Software biete an ○ Hardware ○ Kontakte ○ Chiffre

kleinanzeigen

Biete Software

Über 1300 Spiele und Anwenderprogramme für Ihren C-64 gegen Unkosten abzugeben. Liste mit Preisangabe kostenlos. A. Wittgen, Postlagernd 073861 C, 5060 Bergisch Gladbach 1

Verkaufe original „PROTEXT"-Textverarbeitung für Commodore 128 mit Handbuch für 40,– DM. Tel. 0561/898633.

C16/116/4 PLUS/VC-20/C-64 Fibu, VE/EK, Foto-, Film-, Video-Musikarchive, Lotto, Adress, YSI Tabello, Digitalo, Spielepakete, Kartei, Biblo, KFZ, Denktraining Lernprogramme, Handbücher, Staubschutzhaube ab 9,90 DM. Katalog gegen 80 Pf-Marke.
Computerservice T. Hofstede, A. d. Windmühele 8, 5010 Bergheim 5.

C-64! Nur 1 DM je Programm! C-64! Clubsoftware (Anwender-, Spiel- Lern-, Denk- und Musikprogramme gegen Kopiergebühr von nur 1 DM abzugeben. Z. B. Vokabeltrainer, Astrologieprogr. mit Bildschirm- u. Drukkerausgabe, Synthesizer, Biorhythmus, Actionspiele und Hilfsroutinen. Kein Schund! Info (Rückporto. **TEXASSOFT, Nerthastr. 26, 8 München 19**

★ C16/116 ★ Grafikprogramme ★ Mini-CAD, sichern, laden ★ ★ und weiterbearb. d. Grafi- ★ ★ ken; Info gegen Porto; M. ★ ★ Rätzel, Ulvenbergstr. 6, D- ★ ★ 6100 Darmstadt ★★★★★

Hallo C16 Fans und 116 Plus 4 Ich habe für Euch den Automaten Super 7 programmiert. Cassette + Listing DM 10,– Schein Stichwort S7. Rolf Leesemann, Feldstr. 40, Postfach 3172, 2300 Kiel 1. Lieferung: 5 Tage.

Über 1300 Spiele und Anwenderprogramme für Ihren C-64 gegen Unkosten abzugeben. Liste mit Preisangabe kostenlos. A. Wittgen, Postlagernd 073861 C, 5060 Bergisch Gladbach 1

C-16/C-116 Superspiele! Info gegen Rückporto bei Hannes Kaltenbach, Prielmayerstr. 16, 7990 Friedrichshafen 1.

C-16/116 ★ Plus 4 ★ C-16/116
Software ? – na klar!
- **Grafik**
- **Sound**
- **und für die Schule**

Eine ausführliche Info gibts gegen Rückporto bei:
B. Lauer, Handgasse 10, 8700 Würzburg.

● ● ● ● C-16/C-116 ● ● ● ● Spiele, Anwender-Progr. Softw.-Schutz(Broschüre), Turbo-Tape (3600 Baud) zu verkaufen. Alle Programme auf Cassette. A. Lämmerzahl, Saarstr. 85, 4100 Duisburg 17. Ruf: 02136/113994.

VC-20 Spitzenprogramme für 6V, 8K und 16K. Liste für nur 50 Pfe-Briefmarke bei M. Platt, Carl-Orff-Str. 4, 8721 Hambach! Jetzt preiswert: Zentrax (16K). Ein S.F.-Abenteuer mit herrlichen 3D-Bildern. Nur 180 DM. 3 Actionspiele für die 6V, nur 14 DM! Super Spiele – super billig!

C16/116 ★ C16/116 ★ C16/116 Das erste Gaphic-Adventure „Die Jünger des Todeslichtes" für den C-16/116 ist da. 14 Seq. a 12KB = 168 KB. Preis? Ganze 20,– DM. Bestellen bei: Jörg Kowalski, Bickernstr. 161, 4650 Gelsenkirchen. Oder Info gegen Rückporto. Nur als Cassette lieferbar!!

Für alle Schneider-Besitzer Verkaufe ich viele Spielprogramme. Preisliste u. Beschreibung gegen Briefmarke. Keine Raubkopien! K. Ley, Postfach 2124, 2390 Flensburg.

VC-20 ★ ☆ ★ ☆ ★ ☆ ★ C-64 Neue Liste Spiel + Anwendersoftware. Info bei:
Volker Coert, Söllingstr. 4–8, 4300 Essen 1. Bitte Rückporto!

CPC 464 ★ CPC 664 ★ CPC 6128 Verkaufe super CHEMIEPROGRAMM 10 DM bei Cas., 25 DM bei Disc. Tausche auch Programme. Liste an: Holger Groß, Am Ginsterberg 13, 6630 Saarlouis 4

CPC ● TOP-SOFTWARE ● CPC ● ● Biorhythmus-Total ● ● Läßt vergleichbare Programme erblassen. Exklusiver Bildschirm, doppelte Grafik, Vergrößerung, sehr komfortabler Ausdruck – Ein Monat ca. 8 DIN-A4-Seiten mit sehr ausführlichen Erläuterungen und Hardcopy für Grafiken. Zusätzlich können Partnerschaftsanalysen erstellt werden, ebenfalls mit ausführlichem Ausdruck. Interessante Nebenverdienstmöglichkeiten bietet dieses Programm, das sich auf rein wissenschaftliche Grundlagen und Erfahrungen stützt (49K). Bestellungen unter Nr. 0961 gegen Vorkasse bei Hilterscheid, Ludwig-Rosenberg-Ring 47, 2050 Hamburg-80, Tel. 040/7211974, Cass. DM 30,–, Disk DM 40,–

C-16/C-116/plus 4 Musikcompiler für Interruptgesteuerte Musik; Sonderzeichen für Ihre eigenen Programme; Vokabelprogramm mit französischen Zeichen; Spiele; Lernprogramme etc. Liste anfordern! BILLIG PREISE! Tel. 02581-6715, M. Suer, Walgernweg 22, 4410 Warendorf 1

★ C-64 ★ Spitzensoftware ★
★★★★★★★★★★★★★★★
Adressenverw. (D) 12,95
Videokass.-Verw. (D) 12,95
★ Jed. Prg. auf Disk + Handbuch. Info geg. 80 Pf. Preis per NN + Porto bei: Michael Rau, Hüheweg 46, 8580 Bayreuth.
★★★★★★★★★★★★★★★

C-16/116 ★ C-16/116 ACTION Bei mir gibt es Topspiele für nur 20,– DM. Diesen Monat sind es: PENGO, TERMINATOR, SHAOLIN, BOMBER, DALLAS. Den Bestellungen bitte Schein oder Scheck beilegen. Bestellungen bitte an: Roland Fuchs,- Blanckenagelweg 9, 4770 Soest/Westf. P.S. Tausche auch. Angeben ob Diskette oder Kassette. ★★★★★★

★ C16/116 ★ Grafikprogramme ★ Mini-CAD, sichern, laden ★ ★ und weiterbearb. d. Grafi- ★ ★ ken; Info gegen Porto; M. ★ ★ Rätzel, Ulvenbergstr. 6, D- ★ ★ 6100 Darmstadt ★★★★★

Tausch-Biete C-16/116 Software
C-16/116
C-16/116
C-16/116
Tausche – Biete C-16/C-116 Software. Habe Utilities, Textverarbeitung, Plotterfunktionen, Mathematik, Knobeleien u. a. nützliche (!!!!) Programme: Info bei St. Sedlaczeck, Florastr. 11, 5063 Overath

Biete Hardware

★★★ C-16 Hardware ★★★
★ 16K Ram Erweiterung 89,– ★
★ 64K Ram Erw. (nur C16) 169,– ★
★ Joystickadapter 10,– ★
★★★ C-16 Software ★★★
★ Favourite 4 (Sammlung) 30,– ★
★ Plus Paket (Sammlung) 30,– ★
★ Turbo Tape 15,90 ★
★ Tel. Best. 07307/6230 ★
★ **Comp. Service PF 1304 7913 Senden** ★

★ Commodore Hardware ★
★ Drucker Comrex CR220 349,– ★
★ Drucker Star Gemini 10x 578,– ★
★ Oki 182 798,– ★
★ Schutzhaub. C64/C16 ab 12,90 ★
★ Schutzhauben C128 15,90 ★
★ Joystick Quickshot II 19,90 ★
★ Disketten Select 2D 27,90 ★
★ Tel. Bestellung: 07307/6230 ★
★ **Comp. Service PF 1304 7913 Senden** ★

Für VC-20: Biete 24KB Erweiterungsmodule an. Schaltbar von 3KB bis 24 KByte. Preis: 90,– DM. Informationen gegen Rückporto. Friedel, Heinz, An der Halde 43, 4330 Mülheim-Ruhr.

Commodore C-16 + Datasette + Basic-Lehrbuch und Kassette incl. Garantie bis 30.9.1986 nur DM 189,–!!! ★★★ Commodore C16 Speichererweiterung nur DM 79,– incl. Einbau und Funktionsgarantie auf 60671 Bytes free!! = Selbstkostenpreis. **128 K für C-16** in Kürze!! Info: Tel.: 06201/69189 ★★★ Anwender- u. Spiele-Software vorhanden.

★ **Hallo C-16 Besitzer** ★
★ Wie wäre es, wenn Sie ihr ★
★ C-16 nach dem Einschalten so ★
★ begrüßt ★
★★ **60671 BYTES FREE** ★★
★ Info gegen Freiumschlag ★
★ Markus Antel, Buchenweg 14 ★
★ 8205 Kiefersfelden ★
★★★★★★★★★★★★★★★

ACHTUNG C16/116 Besitzer Speichererweiterung 64K (60671) Bytes free) zum Superpreis von 59, 50 DM + 5 DM f. Porto u. Verpackung. (Muß eingebaut werden, Bauanleitung liegt bei). Martin Brüggemann, Bachstr. 10, 5940 Lennestadt 17, Tel. Ab 16 Uhr 02721/ 82827.

Verkaufe wegen Systemwechsel Commodore C-16 mit Datasette u. Joy-Adap. v. verschiedene Programme. (DM 250) – 5 Monate alt). An Jan Linek, Blumenthalstr. 9, 8500 Nürnberg 80, Tel. 0911/263286.

Verkaufe Schneider CPC 464. Mit grün Monitor und 5 Programmen wegen Systemwechsel zu verkaufen. Preis: VB. Frank Siegel, ☎ 07021/6724

★★★★★★★★★★★★★★
★ **ACHTUNG C16/C116 Be-** ★
★ **sitzer!!** Endlich ist er da, ★
★ der Professionelle Joy- ★
★ stick-Adapter, für nur sage ★
★ und schreibe 9,95 DM und ★
★ wo: Fa. H.P.T. Electronic & ★
★ Co, Am Rain 23–27, 8024 ★
★ Deisenhofen, Tel. 089/ ★
★ 6134120 ★
★ Nur solange Vorrat reicht. ★
★★★★★★★★★★★★★★

Verkaufe Commodore 128 PC, Floppy 1571, Monitor 1901, Drucker MPS 803 komplett 2.000,– Neuwert 3.700,– alles erst 2 Monate alt. Abgabe wegen Bundeswehr. Dazu gratis 2 Joystick + 40 Disketten mit Box. Programme wie Nord Star, Datamat, Protext, Winter Games usw. Markus Jung, Schmachtendorfer Str. 158, 4200 Oberhausen 11, ☎ 0208/ 682367

Verkaufe VC-20, VC-20 + Datasette + 16K. Erw. 9 Kassetten (Superspiele) z. B.: Choplifter, Spuckhaus, Schaelt, Fire Galaxy. **Preis: 260,–** (VHB). Tel. 06543/2984 von 16 bis 20 Uhr.

Verkaufe Commodore CBM 8032, DIN-Tastatur **8250** Dual Floppy, Matrixdrucker Itho 1550 incl. Interface + D-P-**Kabel** super günstig! Schmeding, Sachsenstr. 5, 4980 Bünde, Tel. 05223/10971

★ **VC-20** ★ **SUPER** ★ **C-64** ★
★ VC-20 - Speichererweite- ★
★ rung 32 **KB; Insges,ä.ß 27** ★
★ 5K für **Basic, 11K für** MC- ★
★ Prg's. 1MCL. **3K**-Bereich!!! ★
★ **Der** Preis: nur 70 **DM!** Au- ★
★ Berdem Relais-Interface ★
★ **8xUM** für C-64 oder VC- ★
★ **20!** Mit Resettaster nur 50 ★
★ DM! ★ Melden bei: Sven ★
★ Rakers, Feldweg 10, 4425 ★
★ Billerbeck, Tel. 02543/ ★
★ 4238 ★

★ **C16/C116 Ram-Erweiterung!**

C16/116 60671 Bytes free für DM 97,80 – Alle Ports frei – **keine** Einschränkungen – incl. Umbau ★ Computer ohne **Zubehör** mit 100,– DM (2,20 **für die** Post) **an: Elektronik-Technik, Tannenweg 9, D-2351 Trappenkamp**

Verkaufe Drucker Seikosha GP-100 VC für Commodore 150,– **DM** und **dazu** Eprom **mit** deutschem Zeichensatz auf Sekundäradresse **7 für 30,–** DM. Tel.: 0561/898633.

SUPER ANGEBOT!!! Biete den Schneider CPC 464 und Diskettenlaufwerk **für** 2000,– DM (Festpreis) an. **Mit** Anleitungsbüchern für Computer und Floppy. Geeignet für Anfänger: Meldet euch bei: Oliver Ihns, Rispenweg 21c, 2000 Hamburg 53, Tel. 040/842405. **Eilt!!!!** Zwecks Systemwechsels.

Suche

+ + + + **Suche VC-20** + + + +
+ möglichst mit Cassetten- +
+ laufwerk ohne Joystick. +
+ K. H. Wendelstein, **Her-** +
+ mann-Löns-Straße 31, Tel.: +
+ 07123/14513, 7430 Met- +
+ zingen. +
+ + + + + + + + + + + + + +

Suche für den VC-20 „POLE POSITION, FLUCH DES PHARAO." Außerdem viele Action- und Starspiele. Tausche auch. Biete für original Modul „Pole Position" bis 30 DM.
Brauche auch Anwendungsprogramme, Floppy, Akustikkoppler, Drucker und Videokamera zu vernünftigen Preisen. Tel.: 08459/1536

C-16: Suche Kassetten-Anwenderprogramme **(keine Spiele!), 64 K-Ram realisiert. Suche auch** jegliche **Tips zu** Hardware-**Erweiterungen.** Freundliche Angebote **bitte mit** kurzer Beschreibung **an: Ulrich** Vollmer, **Prökelmoor 1, 2000 Hamburg 71.**

Suche Diskettenlaufwerk für C-16. Tel. 04531/3213, **Arno** Fenske, Berliner **Ring 35, 2060** Bad Oldesloh.

Suche Joysticks oder Adapter + Joysticks, um Commodore C16 bedienen zu können.

Suche wenig gebrauchten Drucker für Commodore C-16/ 116 + 64 und Joysticks für C-16/ 116. Klaus Michel, 3565 Breidenbach.

C-16 ★ C-16 ★ C-16 ★ C-16
Suche ein gebrauchtes, nicht zu teures Diskettenlaufwerk für den Commodore 16.
Angebote bitte an die **Tel.-Nr. A-07758/2505.** Es EILT!!!!!!
Verkaufe **Plotter/Printer 1520/** Neu SCHNELL!!!!!

Club-Nachrichten

Geschenkt ist das nun wirklich schon! Für nur 5,– monatlich könnt **ihr Mitglied im** VIC-20-Club „VIC-Freaks" werden. **Davon** wird die umfangreiche Clubzeitung **mit** massenhaft Listings, **Tips und** Tricks usw. **finanziert.** Zusätzlich erhält jedes Mitglied monatlich 6 Computerprogramme, Überraschungspakete selbstredend. Keinen Schund, sondern harte Actionspiele in Maschinensprache!!! **Jedes Mitglied** hat ebenfalls **Zugriff zu** den circa 800 Computerprogrammen der VC-20-Clubs, und **es** werden täglich **mehr.** Für einen lächerlichen Unkostenbeitrag von 5 DM bekommt ihr ein Informationspaket auf Band, sowie massenhaft Spiel- und Anwenderprogramme!! Erstgenanntes überwiegend!
Also Freaks, stürzt Euch auf **die** Kulis und schreibt an A. Spindler, Moschstr. 10, 6522 Osthofen, Telefon 06242/3390
AUCH NICHTMITGLIEDER KÖNNEN MIT UNS TAUSCHEN!

Der C-64/VC-20-mini-club besteht seit April 1984. Ursprünglich war er für C-64 und VC-20 Anwender gedacht, doch inzwischen haben wir auch C-128 und neuerdings auch C-16 User unter unseren Mitgliedern. Unsere Mitglieder sind **über** die gesamte Bundesrepublik **verteilt.** Kontakte zu anderen Computer-Clubs betreiben wir ebenfalls. Es wäre also auch **schön, wenn** andere Clubs mit uns **Verbindung** aufnehmen würden.
Das **Alter** unserer Mitglieder geht von **12 bis 65** Jahren, ist also breit gefächert. Unser Beitrag **beträgt** 2,50 DM im Monat. Dafür **bekommt** man vorert alle zwei **Monate** unsere Clubzeitschrift **bzw. in** zeitschriftlosen Monaten unsere Clubnachrichten. Eine Mailbox ist in Arbeit. Weitere Themen sind unser Club Funk, in dem man auch mal akustischen Kontakt der anderen Mitglieder mitbekommt. Weitere Fragen oder sonstiges werden beantwortet unter der Nummer 02151/ 735946. Info (kostenlos) erhält man beim C-64/VC-20 mini-club, Postfach 1314 in 415 Krefeld 29.

Verschiedenes

★ **C-64** ★ **Spitzensoftware** ★
★★★★★★★★★★★★★★★★
Adressenverw. (D) **12,95**
Videokass.-Verw. (D) **12,95**
★ Jed. Prg. auf Disk + Handbuch. Info geg. 80 Pf. Preis per NN + Porto bei: Michael Rau, Hüheweg 46, 8580 Bayreuth.
★★★★★★★★★★★★★★★★

C16/116 ★ C16/116 ★ C16/116
Das erste Gaphic-Adventure „Die Jünger des Todeslichtes" für den C-16/116 ist da. 14 Seq. a 12KB = 168 KB. Preis? Ganze 20,– DM. Bestellen bei: Jörg Kowalski, Bickernstr. 161, 4650 Gelsenkirchen. Oder Info gegen Rückporto. Nur als Cassette lieferbar!!

```
NT "Score:";SCORE:LOCATE 1,3:PRINT "Scre
en:";HOEHLE
970 '
980 LOCATE XM,14:PRINT CHR$(230):LOCATE
XM,15:PRINT CHR$(231):LOCATE XM,16:PRINT
 CHR$(232);CHR$(240):LOCATE XM,17:PRINT
CHR$(233):LOCATE XM,18:
990 STEIN=INT(RND*2):STEINO=INT(RND*2):W
URF=INT(RND*2)
1000 IF STEINO=1 THEN LOCATE 38,15:PEN 1
:PRINT CHR$(237):PEN 2:LOCATE 38,16:PRIN
T CHR$(238):PEN 1:LOCATE 38,17:PRINT CHR
$(239) ELSE LOCATE 38,15:PRINT " ":LOCAT
E 38,16:PRINT " ":LOCATE 38,17:PRINT " "
1010 PEN 3:IF steino=1 AND wurf=1 THEN L
OCATE 37,16:PEN 1:PRINT CHR$(229):FOR KO
PF=36 TO 3 STEP -1:LOCATE KOPF,16:PRINT
CHR$(234):CALL &BD18:PRINT CHR$(22);CHR$
(0):SOUND 1,15,1,2:LOCATE 37,16:PRINT "
":IF KOPF=XM AND YM=14 THEN WURF=0:STEIN
O=0:GOTO 1160
1020 PRINT CHR$(22);CHR$(0):PEN 1:IF STE
IN=1 THEN STEINK=XM:FOR STEINP=6 TO 16:L
OCATE STEINK,STEINP:PRINT CHR$(242):CALL
 &BD18:SOUND 4,0,2,6,0,0,30
1030 IF INKEY(75)=0 THEN GOTO 1270
1040 IF INKEY(76)=0 THEN GOTO 1350
1050 IF INKEY(74)=0 THEN GOTO 1310
1060 IF TIME-OT>5700 THEN GOTO 1460
1070 IF XM=PERLE THEN SCORE=SCORE+100:FO
R SOUNTE=1000 TO 1 STEP -15:SOUND 2,SOUN
TE,1,7,0,0,20:NEXT SOUNTE:PERLE=1
1080 IF XM=X OR XM=X1 OR XM=X+1 OR XM=X1
+1 THEN 1480
1090 IF XM=GRUB OR XM=GRUB+1 OR XM=GRUB1
 OR XM=GRUB1+1 OR XM=GRUB+2 OR XM=GRUB1+
2 THEN 1160
1100 IF STEIN=1 AND STEINK=XM AND STEINP
>14 AND STEINP<16 THEN 1160
1110 IF XM=PERLE THEN SCORE=SCORE+100:FO
R SOUNTED=1000 TO 1 STEP -13:SOUND 2,SOU
NTED,1,6,0,0,1:NEXT
1120 IF steinp<17 AND STEIN=1 THEN LOCAT
E STEINK,STEINP:PRINT " ":NEXT STEINP
1130 IF STEIN=0 AND STEINO=1 AND WURF=1
THEN LOCATE KOPF,16:PRINT " ":NEXT KOPF
1140 GOTO 990
1150  'LEBEN VERLOREN
1160 INK 0,0:
1170   FOR WAITED=1 TO 50
1180 BORDER INT(RND*26)
1190 SOUND 1,2000,1,7,0,0,20
1200 NEXT WAITED
1210 BORDER 0:INK 0,ZUF
1220 LEBEN=LEBEN-1:
1230 IF HOEHLE>8 THEN HOEHLE=5:SCHWERT=0
 :SCHWERTP=0
1240 IF LEBEN=0 THEN INK 0,11:INK 1,26:I
NK 2,6:PEN 2:PAPER 0:MODE 0:PRINT:PRINT
"Leider verloren!!!!":PRINT:PEN 1:PRINT
"Score:";score:PRINT:PEN 14:PRINT "Press
 [TAB] to":PRINT:PRINT "restart or 'E' t
o":PRINT:PRINT"end!" ELSE 830
1250 IF INKEY$="" THEN 1250
1260 IF INKEY(68)=0 THEN RUN ELSE a$=LOW
ER$(INKEY$):IF a$="" THEN 1260 ELSE IF a
$="e" THEN CALL &BBFF:CALL &BB4E:END ELS
E 1260
1270 LOCATE XM,14:PRINT " ":LOCATE XM,15
:PRINT " ":LOCATE XM,16:PRINT "  ":LOCAT
E XM,17:PRINT CHR$(244):SOUND 1,0,2,3,0,
0,20
1280 XM=XM+1:LOCATE XM,14:PEN 2:PRINT CH
R$(230):LOCATE XM,15:PEN 1:PRINT CHR$(23
1):LOCATE XM,16:PEN 2:PRINT CHR$(232);CH
R$(240):LOCATE XM,17:PEN 1:PRINT CHR$(23
3):CALL &BD18:LOCATE XM,17:PRINT CHR$(20
1)
1290 IF XM>37 THEN HOEHLE=HOEHLE+1:GOTO
1430
1300 GOTO 1040
1310 LOCATE XM,14:PRINT " ":LOCATE XM,15
:PRINT " ":LOCATE XM,16:PRINT "  ":LOCAT
E XM,17:PRINT CHR$(244):SOUND 1,0,2,3,0,
0,20
1320 XM=XM-1:IF XM<3 THEN HOEHLE=HOEHLE-
1:GOTO 1430
1330 PEN 2:LOCATE XM,14:PRINT CHR$(202):
PEN 1:LOCATE XM,15:PRINT CHR$(203):PEN 2
:LOCATE XM,16:PRINT CHR$(204):LOCATE XM,
17:PEN 1:PRINT CHR$(233):CALL &BD18:LOCA
TE XM,17:PRINT CHR$(201)
1340 GOTO 1060
1350 LOCATE XM,14:PRINT " ":LOCATE XM,15
:PRINT " ":LOCATE XM,16:PRINT "  ":LOCAT
E XM,17:PRINT CHR$(244)
1360 IF XM=SCHWERTP THEN GOTO 1520
1370 XM=XM+5:IF XM>37 THEN HOEHLE=HOEHLE
+1:GOTO 1430
1380 LOCATE XM,10:PRINT CHR$(230):LOCATE
 XM,11:PRINT CHR$(231):LOCATE XM,12:PRIN
T CHR$(232);CHR$(240):LOCATE XM,13:PRINT
 CHR$(245)
1390 FOR SOUNTE=1 TO 1000 STEP 20:SOUND
4,SOUNTE,1,7:NEXT SOUNTE:
1400 LOCATE XM,10:PRINT " ":LOCATE XM,11
:PRINT " ":LOCATE XM,12:PRINT "  ":LOCAT
E XM,13:PRINT " "
1410 LOCATE XM,14:PRINT CHR$(230):LOCATE
 XM,15:PRINT CHR$(231):LOCATE XM,16:PRIN
T CHR$(232);CHR$(240):LOCATE XM,17:PRINT
 CHR$(233):
1420 GOTO 1050
1430 BORDER 6,1:FOR SOUNTED=1000 TO 1 ST
EP -5:INK 0,INT(RND*26):SOUND 2,SOUNTED,
1,7:NEXT SOUNTED:
1440 IF HOEHLE<1 THEN HOEHLE=1
1450 IF HOEHLE<>9 THEN SCHWERTP=0:SCHWER
T=0:GOTO 830 ELSE IF SCORE>800 THEN 1510
 ELSE 50000
1460 '
1470 LOCATE XM,14:PRINT " ":LOCATE XM,15
:PRINT " ":LOCATE XM,16:PRINT "  ":LOCAT
E XM,17:PRINT CHR$(244):LOCATE XM,16:PRI
NT CHR$(248):LOCATE XM,17:PRINT CHR$(249
):FOR WAITED=1 TO 500:NEXT WAITED:GOTO 1
160
1480 PEN 1:LOCATE XM,14:PRINT " ":LOCATE
```

```
 XM,15:PRINT CHR$(253):PEN 3:LOCATE XM,1
6:PRINT CHR$(252);" ":LOCATE XM,17:PRINT
 CHR$(251)
1490 FOR SOUNTED=2000 TO 1000 STEP -20:S
OUND 2,SOUNTED,1,7,0,0,1:NEXT SOUNTED
1500 FOR WAITED=1 TO 500:NEXT WAITED:GOT
O 1160
1510 SCHWERTP=INT(RND*30+7):SCHWERT=1:MO
DE 0:PEN 3:BORDER 24,2:INK 0,1:INK 4,2,2
4:PRINT"      ACHTUNG !!!
":PEN 4:PRINT"Letzte Hoehle !!!":FOR WAI
TED=1 TO 1500:NEXT WAITED:MODE 1:PEN 3:G
OTO 830
1520 IF SCORE<700 THEN HOEHLE=5:GOTO 830
1530 'ALLE HOEHLEN DURCHQUERT
1540 RESTORE 1600
1550 tempo=12
1560 status=1
1570 WHILE status<>0 AND NOT brk
1580 READ status,tone,duration
1590 SOUND status,tone,INT(tempo*duratio
n*0.8334),13
1600 WEND
1610 DATA 49,159,2,42,190,2,28,239,2
1620 DATA 17,190,1,10,239,1,49,179,2,42,
213,2,28,239,2,49,159,4,42,190,4,28,239,
4,49,159,2,42,190,2,28,239,2,49,142,2,42
,179,2,28,239,2,49,159,2,42,190,2,28,239
,2,49,142,2,42,179,2
1630 DATA 28,239,2,49,134,2,42,159,2,28,
201,2,17,159,1,10,201,1,49,150,1,42,179,
1,28,201,1,49,134,4,42,159,4,28,201,4,49
,134,2,42,159,2,28,201,2,49,119,2,42,150
,2,28,201,2,49,134,2,42,159,2
1640 DATA 28,201,2,49,119,2,42,150,2,28,
201,2,49,113,4,42,134,4,28,169,4,49,113,
1,42,134,1,28,169,1,49,134,1,42,169,1,28
,239,1,49,113,2,42,134,2,28,169,2,49,106
,4,42,134,4,28,179,4
1650 DATA 49,106,1,42,134,1,28,179,1,49,
134,1,42,179,1,28,213,1,49,106,2,42,134,
2,28,179,2,49,106,4,42,127,4,28,159,4,49
,106,1,42,127,1,28,159,1,49,127,1,42,159
,1,28,213,1
1660 DATA 49,106,2,42,127,2,28,159,2,49,
100,4,42,127,4,28,169,4,49,100,1,42,127,
1,28,169,1,49,127,1,42,169,1,28,201,1,49
,100,2,42,127,2,28,169,2,49,95,10,42,127
,10,28,150,10,0,0,0
1670 MODE 1:INK 0,1:BORDER 13:PAPER 0:IN
K 1,6:INK 2,18:INK 3,2,1
1680 PEN 3:PRINT"Herzlichen Glueckwunsch
 !!!
":PEN 2:PRINT"Sie haben es geschafft 10
Hoehlen zu
":PRINT"durchqueren."
1690 LOCATE 1,7:PRINT "Sie erreichten":P
EN 1:LOCATE 16,7:PRINT score:PEN 2:LOCAT
E 22,7:PRINT "Punkte.":PEN 1:PRINT "====
=====================================
1700 IF SQ(1)<>4 THEN 1700 ELSE CLEAR
1710 FOR A=1 TO 360
1720 MOVE 300,100
1730 PLOT 300+100*SIN(A),100+90*COS(A),I
NT(RND*4)
1740 DEG:NEXT
1750 MOVE 230,160:DRAW 295,270,2:MOVE 37
6,160:DRAW 305,270
1760 MOVE 250,20:DRAW 295,100:MOVE 350,2
0:DRAW 305,100
1770 LOCATE 19,24:PRINT CHR$(233):LOCATE
 19,23:PRINT CHR$(232);CHR$(240):LOCATE
19,22:PRINT CHR$(231):LOCATE 19,21:PRINT
 CHR$(230)
1780 FOR A=1 TO 360
1790 DEG
1800 MOVE 300,100:
1810 PLOT 300+10*COS(A),100+10*SIN(A),1
1820 NEXT
1830 FOR DUR=1 TO 150:FOR P=1 TO 3:FARB=
(RND*25)+2
1840 INK P,FARB:NEXT P,DUR
1850 LOCATE 1,10:PRINT "Nocheinmal <j/n>
 ?"
1860 A$=UPPER$(INKEY$):IF A$="" THEN 186
0
1870 IF A$<>"J" AND a$<>"N" THEN 1860
1880 IF A$="J" THEN RUN ELSE CALL &BBFF:
CALL &BB4E:MODE 1:END
50000 'ZUWENIG PUNKTE FUER LETZTE HOEHLE
50001 MODE 1:BORDER 6,3:INK 0,2:PEN 1:PR
INT "Ihre Punkte reichen nicht aus,
":PRINT "deshalb koennen Sie ab der 5.
":PRINT "Hoehle weitere Perlen einsammel
n":FOR waited=1 TO 2000:NEXT waited:hoeh
le=5:GOTO 820
```

CHECK V1:PRUEFSUMMEN:

ZEILENNR.:	SUMMEN
10- 20:	13944
30- 40:	40890
50- 60:	33923
70- 80:	45169
90- 100:	33808
110- 120:	40066
130- 140:	43124
150- 160:	25333
170- 180:	54895
190- 200:	43049
210- 220:	64982
230- 240:	27451
250- 260:	39871
270- 280:	64674
290- 300:	30776
310- 320:	22458
330- 340:	78744
350- 360:	48494
370- 380:	81184
390- 400:	16957
410- 420:	22383
430- 440:	70152
450- 460:	20528
470- 480:	22387
490- 500:	48259
510- 520:	40738
530- 540:	52276
550- 560:	59846
570- 580:	26914
590- 600:	78843
610- 620:	55808
630- 640:	77584
650- 660:	38502
670- 680:	56232
690- 700:	60025
710- 720:	21858
730- 740:	37825
750- 760:	55414
770- 780:	65001
790- 800:	67579
810- 820:	8127
830- 840:	84255
850- 860:	65936
870- 880:	35004
890- 900:	46466
910- 920:	111428
930- 940:	50648
950- 960:	63379
970- 980:	62700
990- 1000:	14522
1010- 1020:	29313
1030- 1040:	25541
1050- 1060:	31185
1070- 1080:	78345
1090- 1100:	95034
1110- 1120:	68497
1130- 1140:	31204
1150- 1160:	11543
1170- 1180:	21968
1190- 1200:	11413
1210- 1220:	24245
1230- 1240:	93130
1250- 1260:	13111
1270- 1280:	100254
1290- 1300:	63658
1310- 1320:	108674
1330- 1340:	29145
1350- 1360:	47449
1370- 1380:	59715
1390- 1400:	74933
1410- 1420:	10769
1430- 1440:	46620
1450- 1460:	23158
1470- 1480:	80773
1490- 1500:	57428
1510- 1520:	79011
1530- 1540:	23871
1550- 1560:	14990
1570- 1580:	76469
1590- 1600:	14570
1610- 1620:	34014
1630- 1640:	29920
1650- 1660:	62907
1670- 1680:	71680
1690- 1700:	56202
1710- 1720:	8858
1730- 1740:	15647
1750- 1760:	56713
1770- 1780:	68829
1790- 1800:	1826
1810- 1820:	44292
1830- 1840:	54499
1850- 1860:	57514
1870- 1880:	74957
50000-50001:	69760

GESAMTCHECKSUME: **2061**

sie ist da ...

die neue Generation der Computer-Zeitschriften

- 100 Seiten Programme im Test
- der Software-Markt auf einen Blick
- speziell gesuchte Software kann direkt beim Hersteller bestellt werden
- großer Kleinanzeigenmarkt
- für alle Anwender von Heimcomputern

ASM

Tronic-Verlag GmbH · Am Stad 35 · 3440 Eschwege · Telefon: 0 56 51 / 3 00 11